

Dynamics of the Douglas Rachford Method for Ellipses and p -Spheres

Jonathan M. Borwein	Scott B. Lindstrom
CARMA	CARMA
University of Newcastle	University of Newcastle
Brailey Sims	Matthew P. Skerritt
CARMA	CARMA
University of Newcastle	University of Newcastle

Anna Schneider
Universität der Bundeswehr
München

October 14, 2016

Abstract

We expand upon previous work which examined the behavior of the iterated Douglas-Rachford method for a circle and line by considering two generalizations: that of a line together with a p -sphere and that of a line together with an ellipse. With computer assistance, we discover a beautiful geometry that illustrates phenomena which may affect the behavior of the iterates by slowing or inhibiting convergence for feasible cases. We prove local convergence for the case of an ellipse and a line, and — seeking a better understanding of the behavior — we employ parallelization and the use of GPU programming in order to study behavior graphically. Motivated by the computer-assisted discoveries, we prove a result about behavior of the method in infeasible cases.

1 Introduction

In [6] Borwein and Sims investigated application of the Douglas-Rachford algorithm to a line intersecting a circle in 2 dimensional Euclidean space. Among the partial results obtained was local convergence near each of the intersection points. Based on extensive computer experimentation, they were led to ask whether it could be extended to converge to one or other of the two intersection points for all starting points except those lying on a “singular” line orthogonal to the line in question and passing through the centre of the circle. This question was answered in the affirmative by Benoist [8] who showed the result by constructing a Lyapunov function.

We expand upon this work by considering two generalizations: that of the line with a p -sphere and that of a line with an ellipse.

1.1 Preliminaries

The Douglas-Rachford algorithm was introduced half a century ago in connection with nonlinear heat flow problems to find a common point to two closed constraint sets [12].

Let H be a Hilbert space with inner product $\langle \cdot, \cdot \rangle$ and induced norm $\| \cdot \|$. The projection onto a proximal subset C of H is defined for all $x \in H$ by

$$P_C(x) := \left\{ z \in C : \|x - z\| = \inf_{z' \in C} \|x - z'\| \right\}.$$

When C is closed and convex the projection operator P_C is single valued.

The reflection mapping through the set C is then defined by

$$R_C := 2P_C - I,$$

where I is the identity map of H .

Definition 1. Given two closed sets A and B , and an initial point $x_0 \in H$, the Douglas-Rachford method generates a sequence $(x_n)_{n=1}^\infty$ as follows:

$$x_{n+1} \in T_{A,B}(x_n) \quad \text{where} \quad T_{A,B} := \frac{1}{2}(I + R_B R_A). \quad (1)$$

When the two sets A and B are clear from the context we will simply write T instead of $T_{A,B}$. Figure 1.1 illustrates the construction of one iteration of Douglas-Rachford method $T_{E,L}$ where the sets are an ellipse E and a line L .

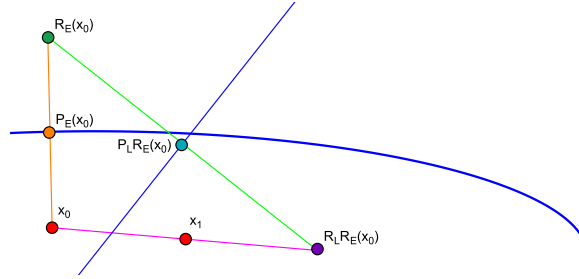


Figure 1: One iteration of Douglas-Rachford method.

The weak convergence of the Douglas-Rachford iteration scheme for closed convex sets with nonempty intersection was proved first by Lions and Mercier in 1979:

Theorem 1 (Lions-Mercier). *Suppose $A, B \subseteq H$ are closed and convex with non-empty intersection. Given $x_0 \in H$ the sequence defined by*

$$x_{n+1} := T_{A,B}x_n \quad \text{where} \quad T_{A,B} := \frac{1}{2}(I + R_B R_A)$$

converges weakly to an $x \in \text{Fix} T_{A,B}$ with $P_A x \in A \cap B$.

Proof. See [14]. □

Despite the absence of satisfactory theoretical justification, the iteration scheme has been used to successfully solve a wide variety of practical problems in which one or both of the constraints are non-convex. Phase retrieval problems are one instance of this. The case of a line and a sphere in Hilbert space is a simple prototype for such situations, which was considered by Borwein and Sims in [6]. Among other results, they established local convergence of the Douglas-Rachford algorithm near each of the intersection points. Benoist extended this result in [8] and proved that the sequence of Douglas-Rachford iterates converges in norm to the point of intersection nearest the initial value except when it belongs to the hyperplane of symmetry.

We investigate the generalization of the Douglas-Rachford iteration scheme for the case of a line L but with the sphere (circle in 2 dimensional Euclidean space) replaced by an ellipse E or a p -sphere S . For the sake of simplicity, we introduce the following notation.

Definition 2. By a suitable rotation and scaling of axes we may without loss of generality write:

$$\begin{aligned} L_{m,\beta} &:= \{(x, y) \in \mathbb{R}^2 | y = mx + \beta\} \\ L_m &:= L_{m,0} \\ E_b &:= \{(x, y) \in \mathbb{R}^2 | x^2 + \left(\frac{y}{b}\right)^2 = 1\} \\ S_p &:= \{(x, y) \in \mathbb{R}^2 | (x)^p + (y)^p = 1\}. \end{aligned}$$

When it is clear from the context what the parameters are we will simply write L , E or S respectively. Similarly, when the context makes it clear we will write T in place of $T_{E,L}$ or $T_{S,L}$.

We have elected to study these two more general kinds of sets in order to gain further insights about the behavior of the Douglas-Rachford algorithm in the case of nonconvex constraint sets prototypical of those emerging in the phase retrieval problem.

1.2 Numerical Methods of Visualization

For the case of the 2-sphere, the projection has a simple closed form. For $(u, v) \neq 0$,

$$P_S((u, v)) = \left(\frac{u}{\|(u, v)\|}, \frac{v}{\|(u, v)\|} \right) = \left(\frac{u}{\sqrt{u^2 + v^2}}, \frac{v}{\sqrt{u^2 + v^2}} \right). \quad (2)$$

This simple closed form is immediately lost for any p -Sphere with $p \neq 2$ or any ellipse with $b \neq 1$ because the induced Lagrangian problem no longer solves cleanly. For this reason, we used numerical methods to analyse the behavior. A description of the optimized function solvers for computing the projections for p -spheres and ellipses is given in Appendix 7, and the scripts themselves are available at <https://carma.newcastle.edu.au/DRmethods/>.

In the discussion which follows, we will make use of the following important definition.

Definition 3. By a *basin of attraction* for a periodic point x of period m , we mean a set S containing x with the property that, for the Douglas Rachford

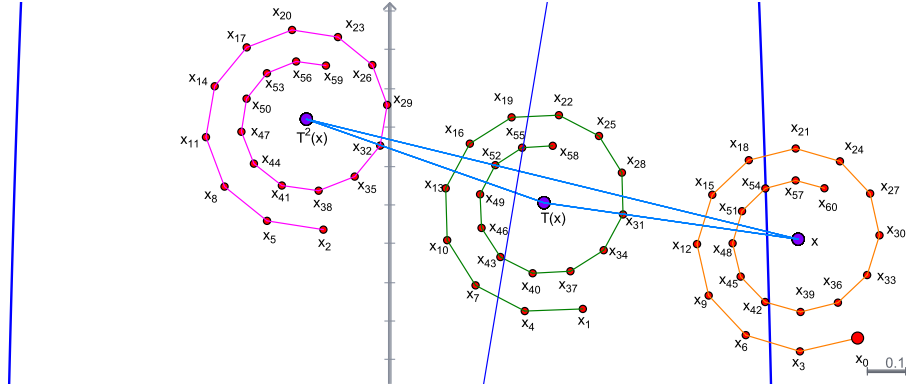


Figure 2: Basins of attraction for periodic points.

method started from any point $x_0 \in S$, it holds that $\lim_{k \rightarrow \infty} x_{km} = x$. In other words, the subsequence of iterates which consists of every m th iterate will converge to x .

Our observations suggest that each of the $m-1$ “offset” subsequences consisting of every $m+k$ th iterate ($k = 1, \dots, m-1$) lie in the basin of attraction for the period m point $T^k(x)$. For an example, see Figure 2. The three large purple points are a period 3 point x together with the two other period 3 points it spawns; namely, $T(x)$ and $T^2(x)$ of the Douglas-Rachford operator $T = T_{E_8, L_6}$. The red points represent the sequence of iterates $x_n = T^n(x_0)$ starting from x_0 . Orange line segments connect successive elements of the subsequence x_{3n} ; green and magenta line segments perform the analogous purpose for the offset subsequences x_{3n+1} and x_{3n+2} respectively. These three subsequences are seen to respectively converge to x , $T(x)$, and $T^2(x)$.

2 Line and p -sphere

2.1 $1/n$ -Spheres for $n \geq 2$ is a Natural Number

For $1/n$ -spheres where $n \geq 2$ is a natural number, period 2 points appear. These have local basins of attraction. Examples of points and their basins of attraction are shown in Figure 3 and Figure 4.

We can begin to describe the fixed points of the square of Douglas-Rachford which are *not* fixed points of the operator itself using the following Proposition about fixed points of T^2 .

Proposition 2. *Let A be any set in \mathbb{R}^2 and $L = L_1$. Let P_A and P_L be the projections onto the sets A and L respectively and R_A, R_L the corresponding*

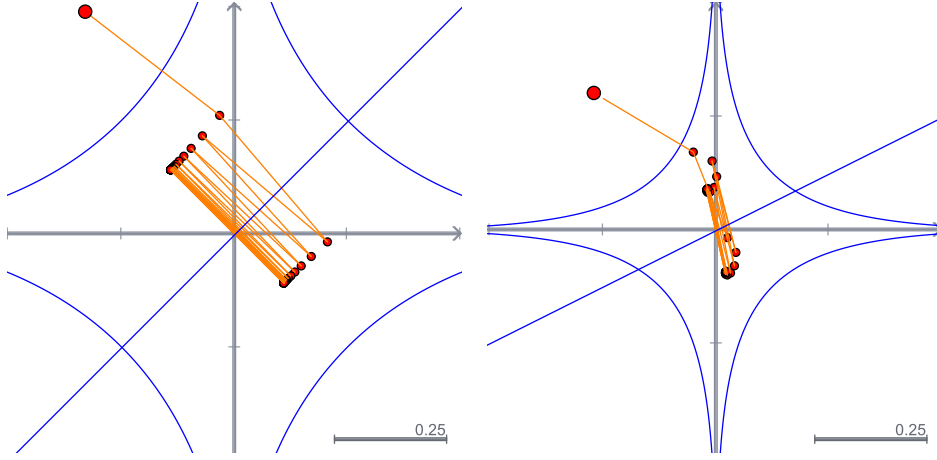


Figure 3: Convergence to period 2 points is illustrated. Left: $T_{S_{1/2}, L_1}$, right: $T_{S_{1/3}, L_{1/2}}$.

reflections. Then $x = (x_1, x_2)$ is a fixed point of T^2 if and only if

$$P_A(T(x)) = \alpha [1, 1] + \beta [1, -1]$$

$$\text{where } \alpha = \frac{1}{2} (x_1 + x_2)$$

$$\beta = \frac{1}{2} \text{sign}(P_A(x)_2 - P_A(x)_1) \cdot |P_A(x)_2 - P_A(x)_1|$$

Here $P_A(x) = (P_A(x)_1, P_A(x)_2)$.

See Claim 10 in Appendix 6 for the proof which follows straight from definitions. Since A is arbitrary this result may be generalized to cover any line L via the application of a suitable translation and rotation.

Corollary 3. For the sphere $S_{1/n}$ with line L_1 , any point $(-t, t)$ or $(t, -t)$ for $t \in (0, \frac{1}{2n}]$ is a period 2 point.

Proof. Simply notice that for any point $x = (-t, t)$, $t \in (0, \frac{1}{2n}]$, we have $\alpha = 0$ and $P_S(x) = (-\frac{1}{2n}, \frac{1}{2n})$. \square

This result, illustrated on the left in Figure 3, provides an analogue of the singular manifold for the case of a 2-sphere. More interesting is the apparent emergence of *basins of attraction* (shown in Figure 4) which have nonzero measure.

These observations already hint at the larger measure and greater complexity of the singular manifold in the case of a line and p -sphere, when $p \neq 2$, compared to that for a line and 2-sphere discussed in [6]. A situation that will be amplified further in what follows.

When we rotate the line to, say, $L_{1/2}$ the period 2 points are no longer constrained to lie in an affine submanifold, and there appear to be only finitely many of them (shown at right in Figure 3). This leads to Question 1.

Question 1. For a sphere $S_{1/n}$ and arbitrary line L_m can we characterize the period 2 points and their basins of attraction? What about points of higher period?

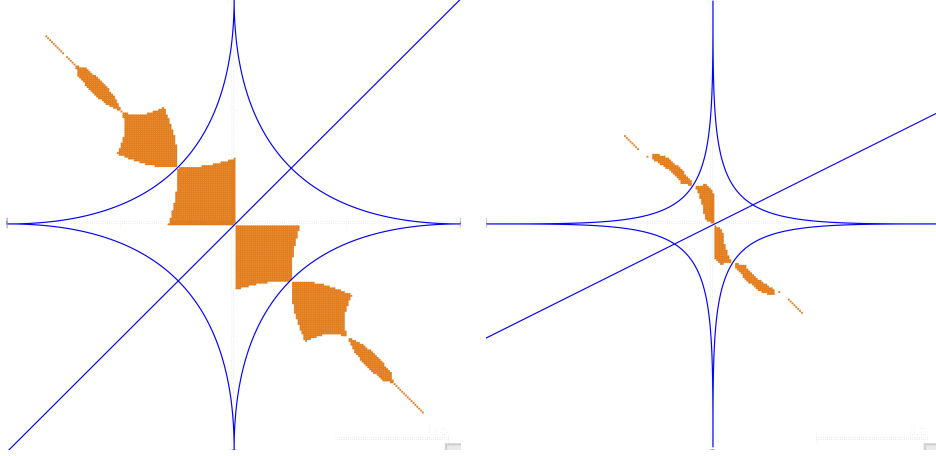


Figure 4: What appear to be basins of attraction for period 2 points are shown in orange. Left: $T_{S_{1/2}, L_1}$. Right: $T_{S_{1/3}, L_{1/2}}$.

2.2 1-spheres

We consider n -spheres for n a natural number. The case $n = 1$ is solvable because all of the constraint equations are of linear form. This allows us to obtain a closed form for many (if not all) period 2 points.

Proposition 4. For the S_1 and L_m where

$$m \in \left(-\frac{3}{2} - \frac{1}{2}\sqrt{17}, -\frac{2}{3+\sqrt{17}} \right) = \left(-\frac{3}{2} - \frac{1}{2}\sqrt{17}, \frac{1}{-\frac{3}{2} - \frac{1}{2}\sqrt{17}} \right) \approx (-3.56, -.280)$$

and $m \neq -1$, the two period 2 points of $T_{S,L}$ are (u, v) where

$$v = -\frac{m^2 + 3m - 2}{5m^2 - 6m + 5} \text{ or } \frac{m^2 + 3m - 2}{5m^2 - 6m + 5} \quad (3)$$

$$u = -\frac{(2m^2 - 3m - 1)}{m^2 + 3m - 2} \cdot v. \quad (4)$$

It is clear from basic geometry that for $m = -1$ any point in the squares $(0, \frac{1}{2}) \times (0, \frac{1}{2})$ and $(-\frac{1}{2}, 0) \times (-\frac{1}{2}, 0)$ are all period 2 points. By symmetry, we similarly have a way of computing the period 2 points for any slope $m \in \left(\frac{3}{2} + \frac{1}{2}\sqrt{17}, \frac{2}{3+\sqrt{17}} \right)$.

Proof. Consider the system

$$u - w = \lambda \cdot 1, \quad v - z = \lambda \cdot 1, \quad |w| + |z| = 1, \quad T(u, v) = (-u, -v). \quad (5)$$

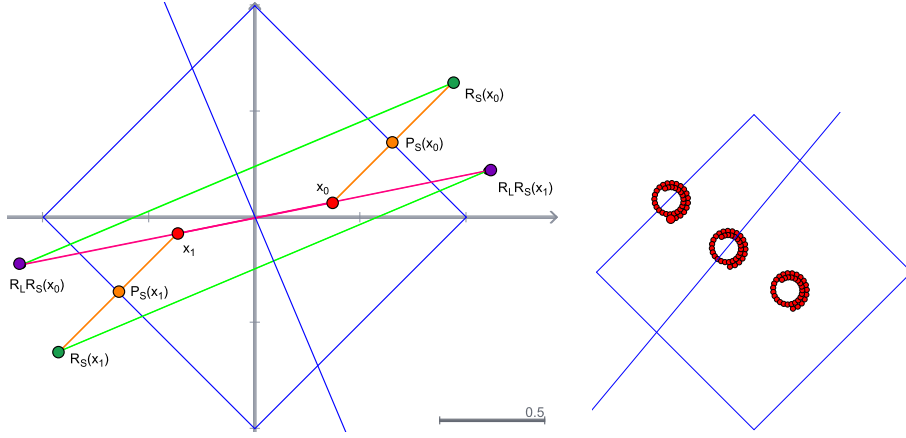


Figure 5: Left: Proposition 4 is illustrated. As the slope of L approaches $-\frac{3+\sqrt{17}}{2}$, we see $R_L R_S(u, v)$ approaches the vertical axis. The line with reciprocal slope $-\frac{3}{2} - \frac{1}{2}\sqrt{17}$ is also drawn for reference. Right: Period 3 points for a 1-sphere.

Where $P_S(u, v) = (w, z)$ the first three equalities are the Lagrangian problem for P_S , and the last two results are true from symmetry. The proof amounts to solving the system in Equation (5). For an arbitrary line $L_{m,b}$, the coordinate $P_L(R_S(u, v))$ is computed by finding the value t for which

$$h(t) = ||R_S(u, v) - (t, m \cdot t) + b||^2 \quad (6)$$

is minimized. Where $R_S(u, v) = (r, s)$, the solution,

$$t = \frac{ms + r - bm}{m^2 + 1} \quad (7)$$

is the horizontal component of the projection. The vertical component is just $m \cdot t + b$. Solving the system in Equation (5) with $b = 0$ and the assumption $\text{sign}(u) = \text{sign}(v)$, we obtain Equations (3) and (4).

It can be seen that Equation (4) is undefined for $m = -\frac{3}{2} - \frac{1}{2}\sqrt{17}$ and has a root at $\frac{1}{-\frac{3}{2} - \frac{1}{2}\sqrt{17}}$ beyond which the signs of u, v will no longer agree. \square

That the boundaries of possible slopes are reciprocals of each other makes sense given the symmetry of the sets. This is illustrated on the right in Figure 5. This leads to the following conjecture.

Question 2. We conjecture that there are no period 2 points for T_{S_1, L_m} for any $m \notin \left(-\frac{3}{2} - \frac{1}{2}\sqrt{17}, -\frac{2}{3+\sqrt{17}}\right) \cup \left(\frac{3}{2} - \frac{1}{2}\sqrt{17}, \frac{2}{3+\sqrt{17}}\right)$.

The ease with which we attain this result in the case of an arbitrary line through the origin motivates us to ask if we can get an even more general result, one for a line with arbitrary intercept b . The introduction of an extra variable means that the category of lines which yield period 2 points may no longer be expressed as simply, since different intercepts correspond to different

sets of viable slopes. We can, however, still obtain a large class of results about period two points for lines with nontrivial intercepts. In the process, we obtain a way to test whether or not a certain class of period 2 points is ruled out for a given line.

Lemma 5. *For $T_{S_1, L_{m,b}}$ where $m \neq -1$, a period 2 point (u_0, v_0) in Quadrant I such that $T(u_0, v_0)$ is in Quadrant III satisfies the following:*

$$u_0 = -\frac{5bm^2 - 2m^3 - 6bm + m^2 + 5b + 4m + 1}{(m+1)(5m^2 - 6m + 5)} \quad (8)$$

$$v_0 = \frac{5bm^2 - m^3 - 6bm - 4m^2 + 5b - m + 2}{5m^3 - m^2 - m + 5} \quad (9)$$

By symmetry of the p -sphere, we obtain a class of results about period 2 points in any quadrant.

Proof. The computation of the projection of onto the line is similar to in Proposition 4. From the Lagrangian problem where $P_S(u_0, v_0) = (w_0, z_0)$ we obtain several equations

$$u_0 - w_0 = \lambda_0 \cdot 1, \quad v_0 - z_0 = \lambda_0 \cdot 1, \quad w_0 + z_0 = 1. \quad (10)$$

We can no longer use the two equations we had previously from symmetry. However, we do obtain a second set of Lagrangian equations. Letting $T(u_0, v_0) = (u_1, v_1)$ and $P_S((u_1, v_1)) = (w_1, z_1)$, we have

$$u_1 - w_1 = \lambda_1 \cdot 1, \quad v_1 - z_1 = \lambda_1 \cdot 1, \quad w_1 + z_1 = -1. \quad (11)$$

Finally, from the fact that $T((u_1, v_1)) = (u_0, v_0)$, we obtain two more equations. Where $T((u_1, v_1)) = (u_2, v_2)$,

$$u_2 = u_0, v_2 = v_0$$

Solving the system yields the result. \square

We also obtain an easy way to check whether a particular line $L_{m,b}$ might yield a period 2 pair in Quadrants I,III for the $p = 1$ sphere. In particular, the answer is no if a particular combination of m, b forces u or v to be negative in Equations (8) and (9).

Question 3. Based on the geometry we conjecture that any period 2 point x_0 for the 1-sphere with an arbitrary line satisfies that $T(x_0)$ is in the opposite quadrant.

If we can answer Question 3 to the affirmative, we will have obtained a way of testing any line of the form $L_{m,b}$ for the existence of possible period 2 points for the 1-sphere.

Graphical analysis suggests that, while more results may be possible, we may not be able to fully describe all of the different possible cases of behavior. For certain lines, points of period greater than 2 appear as in the left of Figure 5 where the line $L_{\frac{6}{5}, \frac{183}{500}}$ yields period 3 points. These can be computed exactly:

$$\left(-\frac{7084639}{13213300}, \frac{6104261}{13213300}\right), \left(-\frac{2265639}{13213300}, \frac{2090461}{13213300}\right), \left(\frac{2875761}{13213300}, -\frac{1455939}{13213300}\right).$$

2.3 n -spheres for $n > 2$ a natural number

In the case $n = 2$, we of course recover the circle for which the convergence properties are known. The case $n > 2$ suggests a conjecture: that for the line not horizontal or vertical there are at most finitely many period 2 points (the $p=3$ image suggests the existence of some) for the p -sphere with $p > 2$. Images

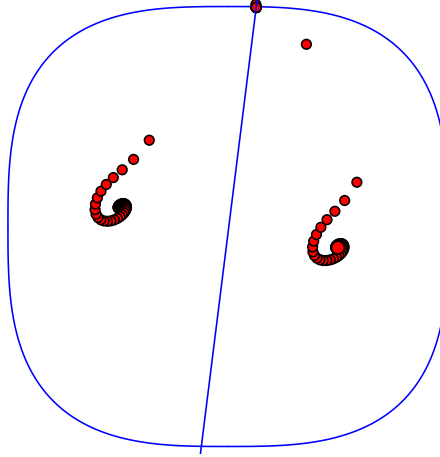


Figure 6: For T_{S_3, L_8} the outward spiraling seen before convergence to a fixed point suggests the source of the spirals may be a pair of unstable period 2 points.

like Figure 6 suggest the likely existence of period 2 points for p -spheres with $p > 2$ and with certain non-horizontal lines, but the points do not appear to be stable.

3 Ellipses

The situation becomes even more interesting in the case of the Ellipse. As the Ellipse is stretched (as b grows), we observe that the upper bound on periods for attractive periodic points appears to grow. The actual appearance of these points is still subject to small perturbations of the line. For the sake of simplicity, we set the intercept at 0 for our pictures and tables, but it should be noted that similar behavior can be observed with nonzero intercepts (although the symmetry between the top and bottom of the ellipse is, of course, lost in this case).

For lines of small slope, there appear to be fewer basins and the periodicities are lower. As the slope is increased, more basins appear, including basins for points of greater periodicity. As the slope of the line continues to increase even further, basins emerge of even greater periodicity, but some of the previously-existing basins appear to shrink in size, eventually becoming repelling rather than attractive. As the line becomes even steeper, all basins eventually appear to become repelling. With even greater slopes, the basins become visually indistinguishable.

This phenomenon is accutely visible in Table 3. It also appears to persist for ellipses stretched even further, although it becomes much more difficult to

	E_2	E_3	E_4	E_5	E_6
L_1	2	2	2	2	2
L_2		2,3	2,3	2,3	2,3
L_3		2,3	2,3,5	2,3,4,5,7	2,3,4,5,7
L_4			2,3,5,7	2,3,4,5,7,9	2,3,4,5,7,9
L_5			2,3	2,3,4,5,7,9,11,13	2,3,4,5,7,9,11,13
L_6				2,3,5,7	2,4,5,7,9,11,13,15
L_7				3	2,3,4,5,7,9,11,13
L_8					2,3,5

Figure 7: Periods observed for attractive basins for various ellipse and line configurations. The notation $p \times n$ indicates that n different sets of basins with periodicity p were observed.

observe accurately because the region of interest becomes far more “busy.” The sensitivity to small perturbations of the line slope is also clear in Figure 12.

In Figure 8, we show for T_{E_8, L_6} the periodic points and, roughly, their basins of attraction. We do so by using thirteen separate sequences of iterates, each starting from a different location. Iterates are colored according to which sequence they belong to. For example, the two large clusters of red dots at the bottom are the iterates of the Douglas Rachford algorithm stemming from a single starting point fairly close to the origin. From top to bottom, the colored iterates show: red-period 6, spring green-period 5, magenta-period 4, chartreuse-period 7, blue-period 3, orange-period 8, cyan-period 5, rose-period 7, green-period 9, violet-period 11, yellow-period 13, azure-period 15, red-period 2.

The colors were chosen by taking a Red-Green-Blue color wheel which included the secondary and tertiary colors (12 colors total) and counting 5 steps around the wheel from one color to the next. The choice of a prime number close to one third of twelve ensures that we paint iterates in contrasting colors while not repeating until the very end.

Our success with the 1-sphere leads us to ask if we can efficiently compute the period 2 points for a general ellipse and line through the origin. In fact, we can, though the closed form is more complicated.

Proposition 6. *For T_{E_b, L_m} we can describe the period 2 points with a closed form which is quick to evaluate. This closed form is in Remark 3 in Appendix 6.*

Proof. Letting $x_0 = (u_0, v_0)$, $P_E(x_0) = (w_0, z_0)$, $Tx_0 = x_1$, $x_1 = (u_1, v_1)$, $P_E(x_1) = (w_1, z_1)$ as in Figure 3. We obtain the system

$$u_0 - w_0 = \lambda 2x, \quad v_0 - z_0 = \lambda \frac{2}{b^2} z_0, \quad u_1 = -u_0, \quad v_1 = -v_0, \quad (12)$$

$$w_0^2 + \frac{z_0^2}{b^2} - 1 = 0. \quad (13)$$

The first two equalities in (12) and the final equality (13) come from the Lagrangian problem while the latter two equalities in (12) come from symmetry. We compute the projection onto the line as we did in the proof of Proposition 4. In order for *Maple* to be able to compute a clean closed form, we must solve in the precise order:

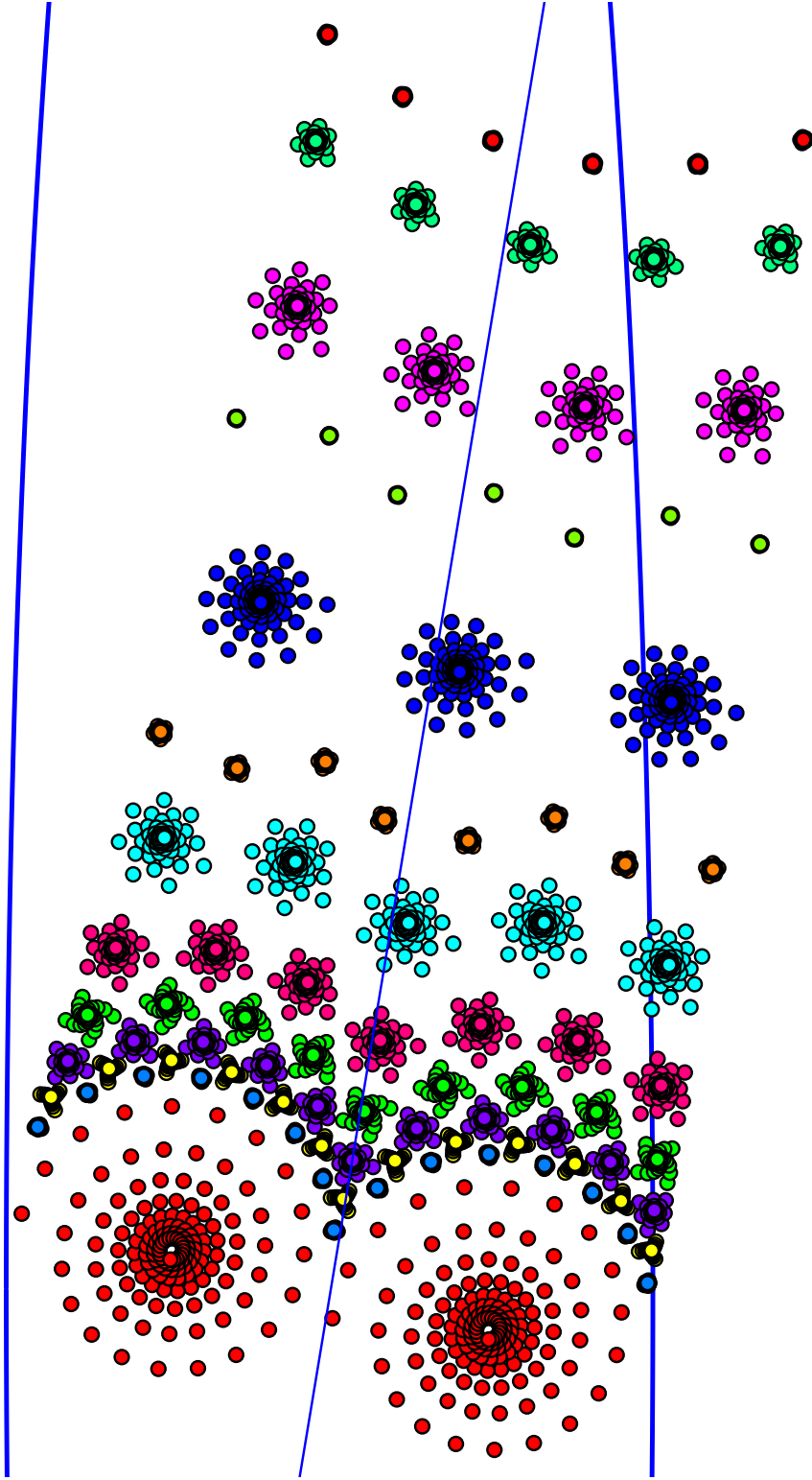


Figure 8: For T_{E_8, L_6} , many points of periodicity appear with corresponding basins of attraction.

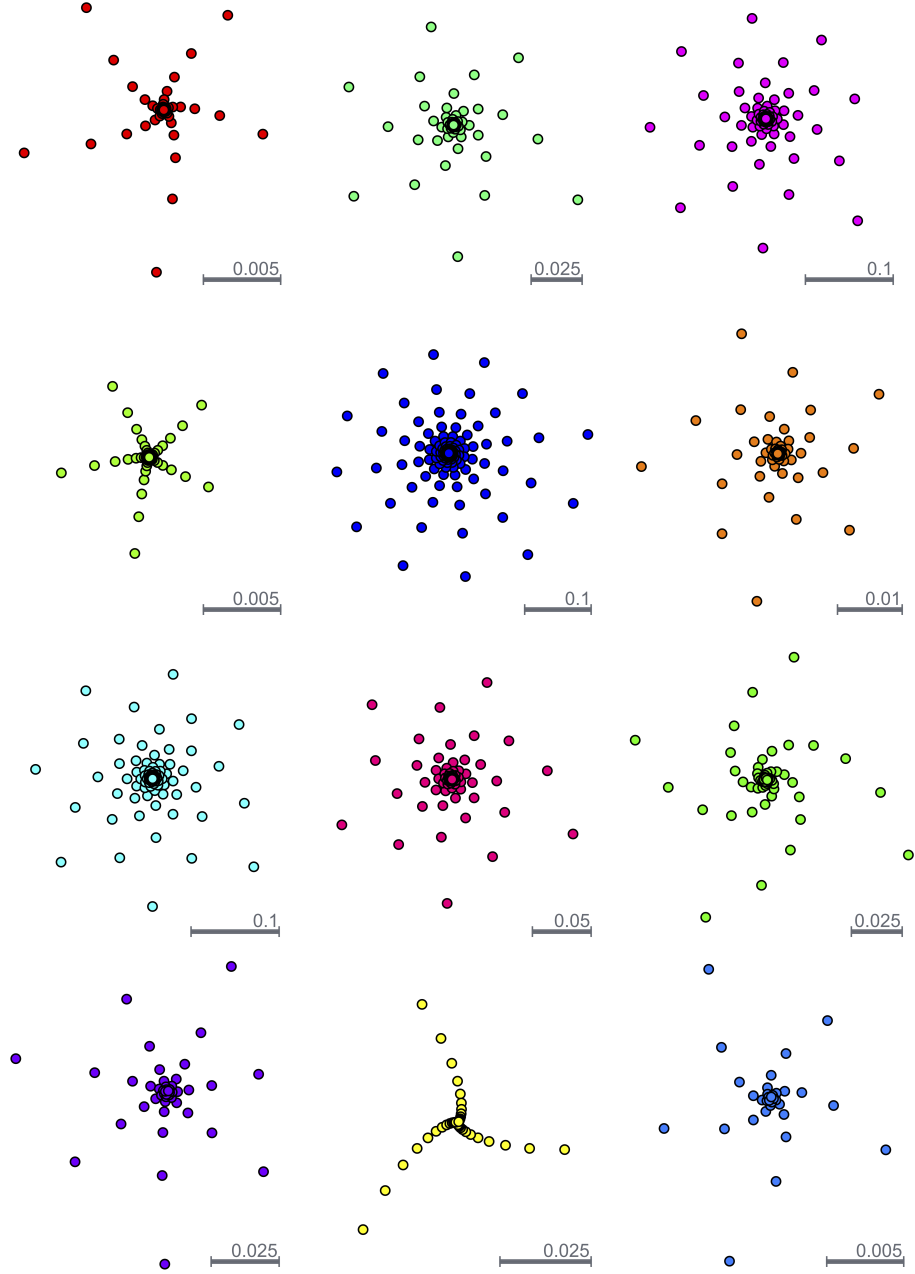


Figure 9: For $T_{E8, L6}$, one spiral is shown from each of the sets of periodic points except for the period 2 points (which are already easily visible in Figure 8). The colors are exactly as they were in Figure 8. $0.32 \times 345.0\text{pt}$

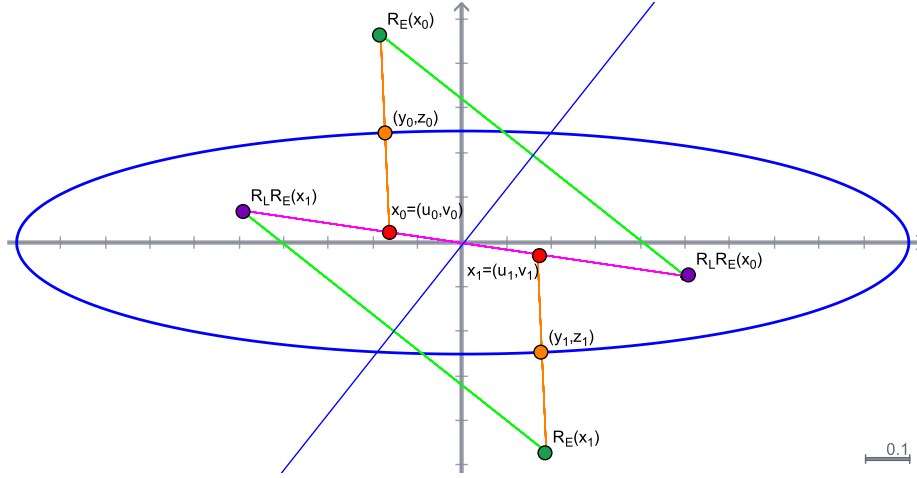


Figure 10: Period 2 points for an ellipse and a line through the origin.

$$\begin{array}{llll}
 1 & y_0 & \text{in} & u_1 = -u_0 \\
 2 & z_0 & \text{in} & v_1 = -v_0 \\
 3 & \lambda & \text{in} & u_0 - y_0 = \lambda 2y_0 \\
 4 & v_0 & \text{in} & v_0 - z_0 = \lambda \frac{2}{b^2} z_0 \\
 5 & u_0 & \text{in} & y_0^2 + \frac{z_0^2}{b^2} - 1 = 0
 \end{array}$$

Maple yields two possible solutions for v . Checking, we find that the second possible solution - the one stated in Equation (31) - is the correct one, consistently matching the period 2 points we observe using *Cinderella*. Choosing that solution for v and solving for u , we obtain the result. \square

Proving the existence of period 2 points algebraically is useful for corroborating that some of the behavior we are seeing in *Cinderella* is, in fact, as it appears to be. However, the degree of complication even for the closed forms of points for the smallest non-fixed period suggested to us that fully describing all behavior globally with explicit forms would be an impractical undertaking. This, in part, led us to pursue the computer assisted evidence-gathering approach we describe in subsections 3.1.

3.1 Studying Convergence: Numerical Motivations

We turn our attention to the study of the basins of attraction for periodic points and local convergence to feasible points. The intersection of an ellipse and a line is locally similar to the intersection of a sphere and a line (at an infinitesimal level, a line with another line). For this reason one would reasonably suspect that convergence of the algorithm should continue to hold locally for the ellipse. However, Benoist's explicit Lyapunov function for the 2-sphere in [8] was sufficiently complicated to suggest that - for the more complicated case of the ellipse - we pursue instead a numerical approximant of the level curves. This approach proved unsuccessful; the reasons why are given in Appendix 8.

For this reason, we sought an alternative method for plotting the curves. This led us to parallelization, the employment of which provided not only a way of plotting curves locally but also a method for clearly visualizing the basins of attraction.

In the absence of an explicit form or numerical approximant for Lyapunov curves, our best method for understanding the behavior in the case of ellipses with $b \neq 2$ is to simulate the behavior with a large number of points and see what happens. Indeed, this is what Borwein did for the case of the sphere and the line prior to Benoist's proof of the Lyapunov function [5, Figure 34]. The results given by such methods are quite compelling: they do not merely motivate the search for a formal proof where one may be attainable, but they also are increasingly necessary in cases where explicit proof may not be possible because the degree of complexity for a given problem precludes it.

"A heavy warning used to be given [by lecturers] that pictures are not rigorous; this has never had its bluff called and has permanently frightened its victims into playing for safety. Some pictures, of course, are not rigorous, but I should say most are (and I use them whenever possible myself)." (J. E. Littlewood, 1885-1977) From Littlewood's Miscellany (p. 35 in 1953 edition), said long before the current graphic, visualization and geometric tools were available.

However, the method we used for Figure 8, though useful for discovery, is not alone sufficient for fully understanding the behavior, even for one specific ellipse and line. There are several reasons for this.

1. There may be other periodic points we cannot see because they are unstable and/or their basins are too small. A case of unstable periodic points for the ellipse is illustrated in Figure 11. Figure 11 also illustrates that slight rotations of the line will move periodic points and may alter their stability.
2. This method of visualization may be deceptive, as it precludes us from really seeing the shapes of the basins accurately. Notice how the patterns of the iterates in Figures 11 and 12 form orderly spirals which look like twisting galaxies. This lovely pattern seems to hold for all the cases we have looked at. For example, zoomed-in pictures of the spirals in Figure 8 are shown in Figure 9. They are certainly beautiful to look at. Intuition would suggest to us that this is perhaps indicative of smooth boundaries for the basins. However, a good visualization of the basins requires a more rigorous approach.
3. The potential for numerical error is accentuated by the fact that the projection onto the ellipse is calculated by solving a function which, depending upon the location of the point for which we are computing, may have an incorrect root quite near to the root we seek.

We sought a clearer picture using parallelization and higher precision.

3.2 Visualization through Parallelization

We wrote a parallelized version of our code to run on a GPU. The code takes each pixel for a given resolution and view and runs 1,000 iterations of T_{E_8, L_6} ,

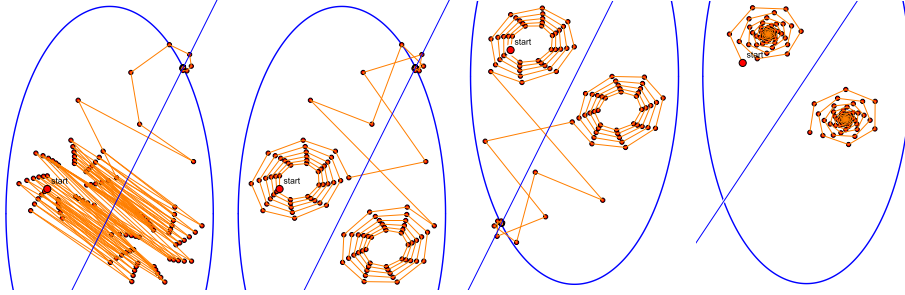


Figure 11: Far Left: T_{E_2, L_2} yields period two points which are unstable. Top Center Left: we alter the code to connect every second iterate, more clearly illustrating the spiraling behavior. Center Right: a tiny perturbation of the starting point is all that determines whether the iterates eventually converge to the “northern” or “southern” feasible point. Far Right: rotating the line to $L_{3/2}$, our periodic points also rotate and become stable instead of unstable.

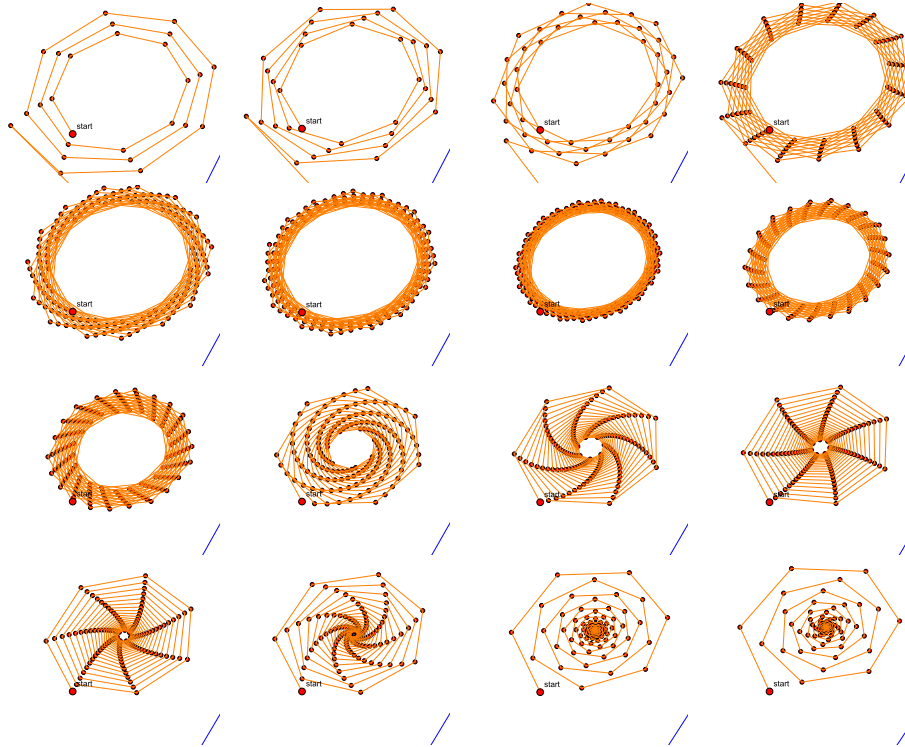


Figure 12: Plotting every second iterate for T_{E_2, L_m} with 300 iterates, we take the line L_2 (starting at top left) and slowly rotate it until we have the line $L_{3/2}$ (finishing at bottom right). Part of the line is visible in the bottom right corner of each frame. As we rotate the line, we see the “speed” at which iterates escape from the source basin decreases until eventually the source basin turns into a sink basin.

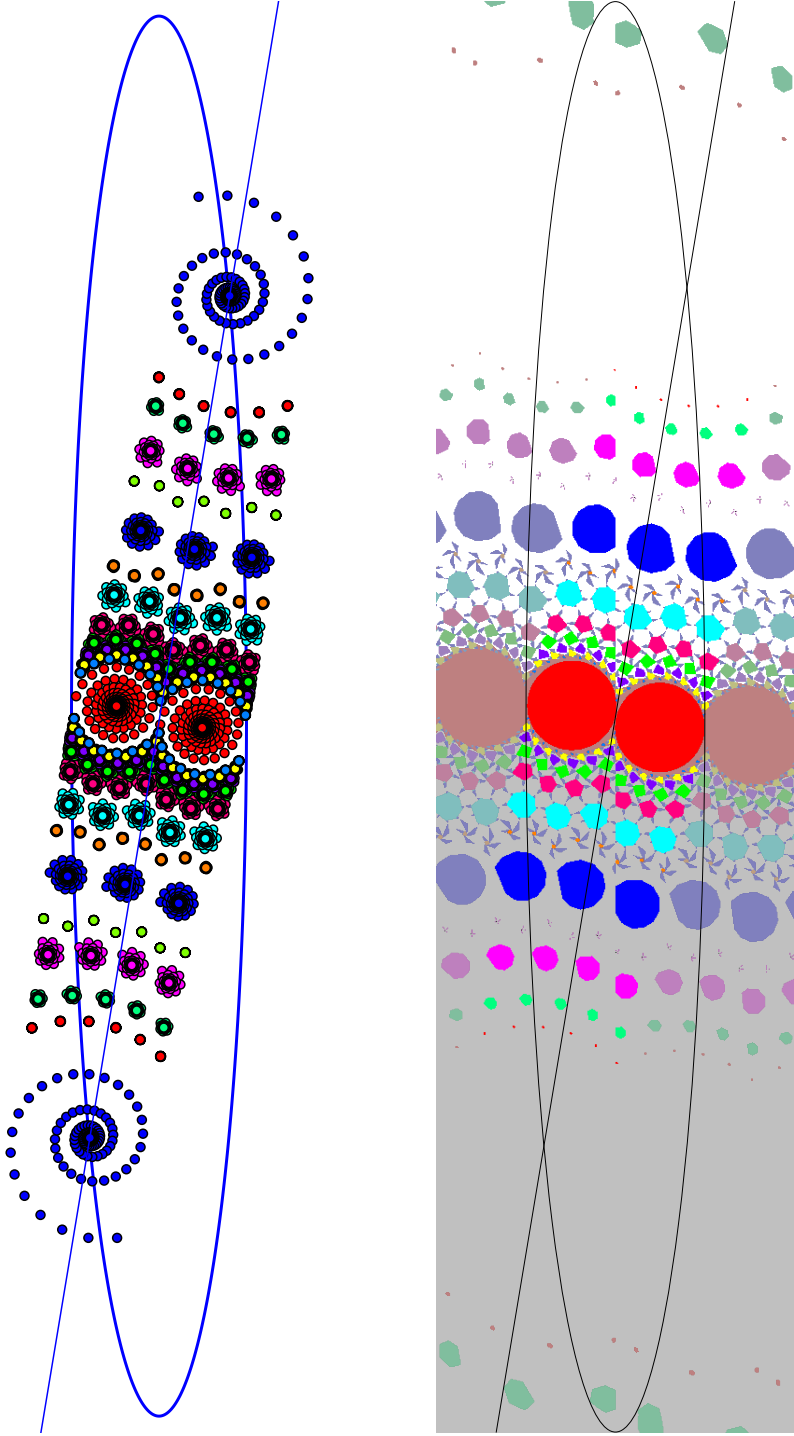


Figure 13: Left: sequences of iterates for T_{E_8, L_6} started at different places in \mathbb{R}^2 (Compare to Figure 8 for a closer zoom). Right: parallelized code shows the basins more clearly (Compare to Figure 14 for a closer zoom).

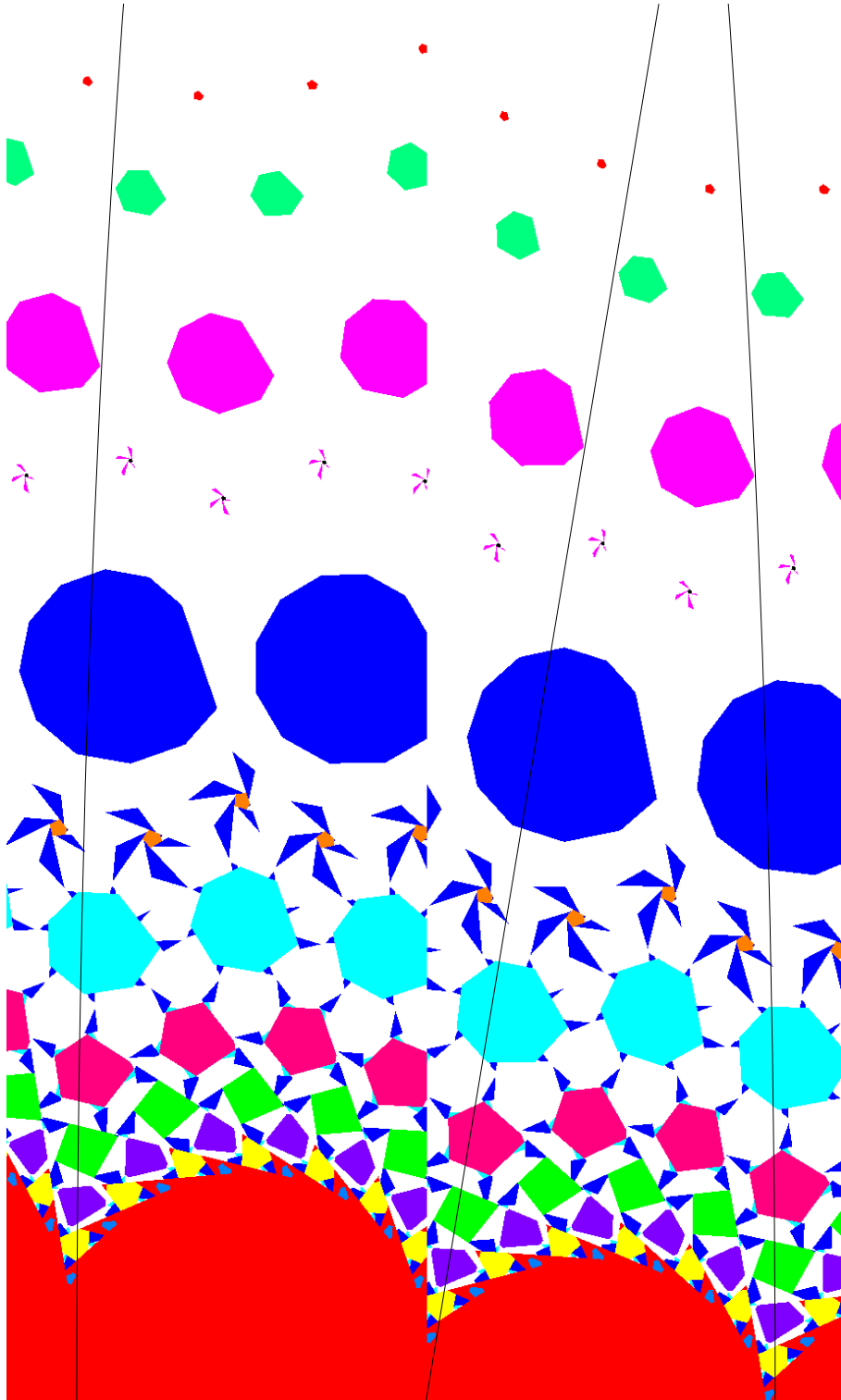


Figure 14: Basins of attraction for T_{E_8, L_6} . See Figure 13 for comparison with plot of iterates.

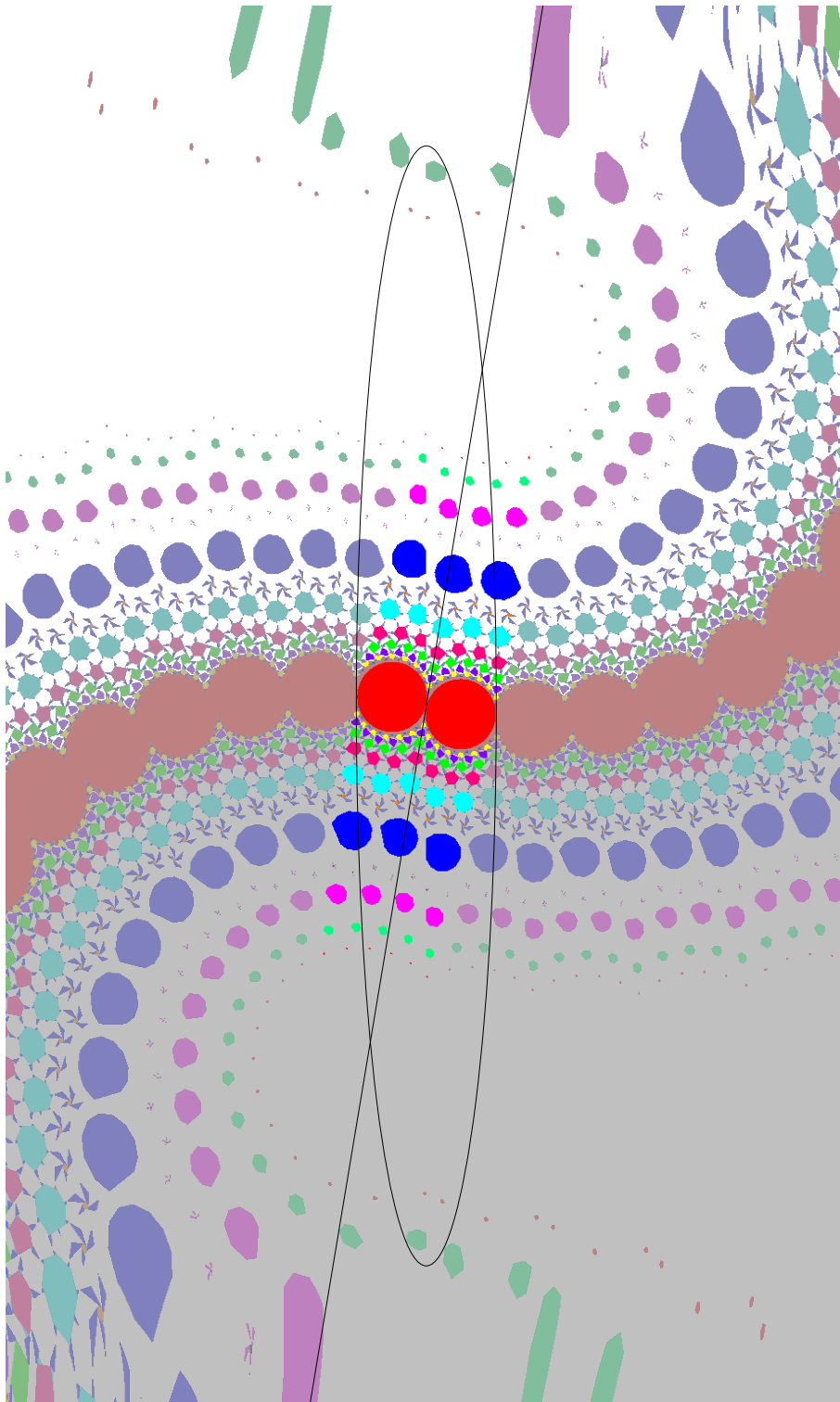


Figure 15: Basins of attraction for T_{E_8, L_6} .

checking the location of the 1,000th iterate against a list containing the feasible points and approximate periodic points. The pixel is then colored according to which point from that list the 1,000th iterate is nearest to.

For Figures 15, 14, and the right image in Figure 13, we chose the colors for assignment based on Figure 8 and the left image in Figure 13. If the 1,000th iterate for a pixel was nearest to a particular periodic point, it was colored the same as the subsequence which converged to that particular periodic point in Figure 8 and the left image in Figure 13. If the 1,000th iterate was nearest to the northern (resp southern) feasible point, it was colored white (resp grey). This allows for an easy comparison. In Figure 13, basins are presented side-by-side with sequences of iterates being pulled into them. In Figure 14, the basins are zoomed in upon and shown in greater detail. In Figure 15, more of the global behavior is visible.

Figure 16 is similarly constructed but colored only according to the two feasible points: points at the center of the source basins whose colors deviate are points for which even the 1,000th iterates have not escaped the repelling basins.

We also created an “artistic” rendering of the image where the colors were chosen based on indigenous Australian tiling art. Figure 24 is in Appendix 9, and further information about the code is given in section 7.3 in Appendix 7.

4 A Theoretical Interlude: Local Convergence for the Ellipse and Line

In [6] Borwein and Sims used the Perron theorem on the stability of almost linear difference equations [13, Corollary 4.7.2] to establish local convergence of the Douglas-Rachford algorithm, $x_{n+1} = T_{K,L}(x_n)$, to an isolated point $f \in L \cap K$ when L is a line and K is the (non convex) unit sphere in a n -dimensional Euclidean space. We outline a strategy for extending this to the case when L is still a line, but K is a smooth hypersurface ($(n-1)$ -manifold). We consider its application when L and K lie in \mathbb{R}^2 ; L is the line $\alpha x + \beta y = \gamma$, K is the ellipse $E: x^2 + \left(\frac{y}{b}\right)^2 = 1$.

THE STRATEGY is to show that in a neighbourhood of f the reflection in the supporting hyper-plane H_f to K at f provides an o -order approximation to the reflection in K so that the Perron theorem can be applied to the system of difference equations corresponding to the Douglas - Rachford algorithm. That is, we want

$$R_K(p) = R_{H_f}(p) + \Delta, \quad \text{where } \|\Delta\| = o(\|p - f\|) \text{ for } p \text{ sufficiently near } f.$$

For the Euclidean reflection this follows if $\|P_K(p) - P_{H_f}(p)\| = o(\|p - f\|)$.

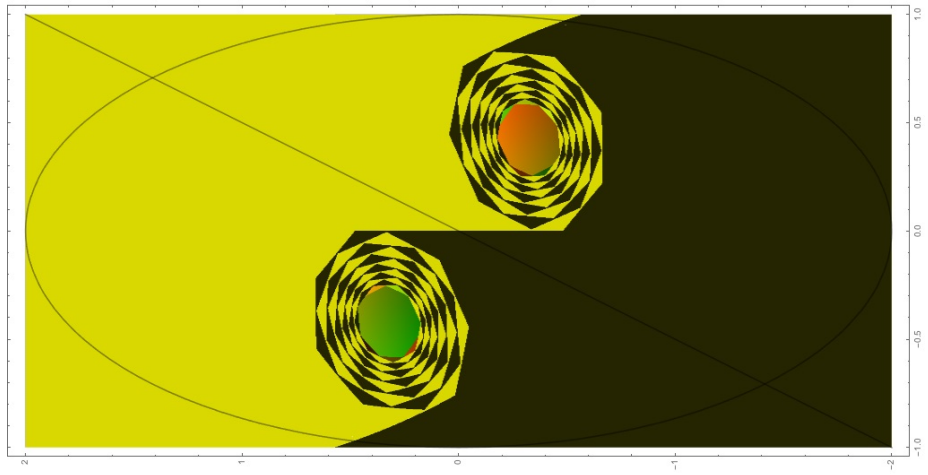


Figure 16: Matt. We want one of these ones here. The $b=2$ ellipse and line $y=2x$.

When this happens we have, for p in a neighbourhood of f ,

$$\begin{aligned}
T_{KL}(p) &= \frac{1}{2} [p + R_L(R_K(p))] \\
&= \frac{1}{2} [p + R_L(R_{H_f}(p) + \Delta)] \\
&= \frac{1}{2} [p + R_{L-f}(R_{H_f}(p) + \Delta - f) + f] \\
&= \frac{1}{2} [p + R_{L-f}((R_{H_f-f}(p - f) + f) + \Delta - f) + f] \\
&= \frac{1}{2} [p + R_{L-f}(R_{H_f-f}(p - f)) + R_{L-f}(\Delta) + f], \quad \text{since } R_{L-f} \text{ is linear} \\
&= \frac{1}{2} [(p - f) + R_{L-f}(R_{H_f-f}(p - f))] + \frac{1}{2} R_{L-f}(\Delta) + f
\end{aligned}$$

So,

$$T_{KL}(p) = f + T_{(H_f-f)(L-f)}(p - f) + \Delta'$$

where $\Delta' = \frac{1}{2} R_{L-f}(\Delta)$ has $\|\Delta'\| = o\|p - f\|$, since R_{L-f} is a bounded linear operator.

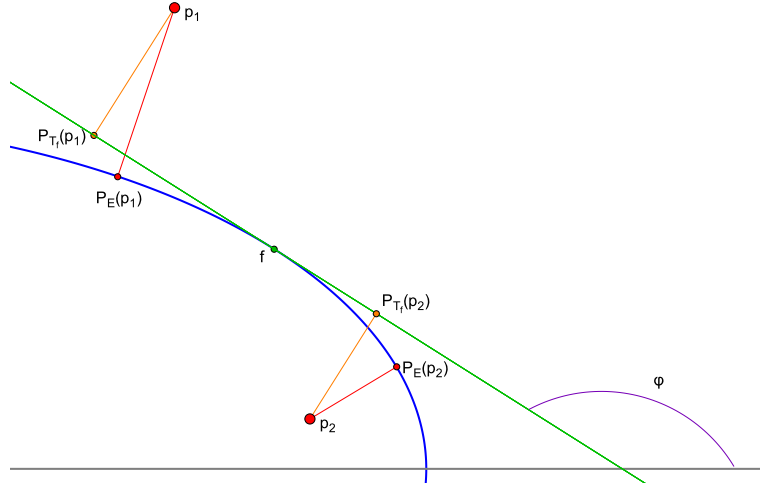


Figure 17: Approximation of P_E by P_{H_f} near f .

Thus, by the theorem of Perron [see, [6] theorem 6.1 or [13] Corollary 4.7.2], the system of difference equations corresponding to the Douglas - Rachford algorithm for K and L ,

$$x_{n+1} = T_{KL}(x_n)$$

is exponentially asymptotically stable at f (in particular $\|x_n - f\| \rightarrow 0$ for x_0 sufficiently near f) provided all the eigenvalues of the linear operator $T_{(H_f-f)(L-f)}$ have moduli less than one.

Remark 1. When M is a subspace, the projection P_M is linear (as is the case

when $M = L - f$) and the Douglas - Rachford operator for N and M becomes

$$T_{NM} = \frac{1}{2} [I + (2P_M - I)(2P_N - I)] \quad (14)$$

$$= 2P_M P_N - P_M - P_N + I \quad (15)$$

$$= P_M P_N + (I - P_M)(I - P_N). \quad (16)$$

When N is also a subspace (for instance when $N = H_f - f$) this may be written as

$$T_{NM} = P_M P_N + P_{M^\perp} P_{N^\perp}$$

where $^\perp$ denotes the orthogonal complement.

ASIDE: As a curiosity we observe that if in the case of two subspaces we define a *twisted Douglas - Rachford operator* by $V_{NM} := P_M P_N + P_{N^\perp} P_{M^\perp}$. Then, since $P_M P_{M^\perp} = P_N P_{N^\perp} = 0$, the iterates $x_n = V_{NM}^n(x_0) = u_n + v_n$, where $u_{n+1} = P_M P_N(u_n)$ and $v_{n+1} = P_{N^\perp} P_{M^\perp}(v_n)$. So, the sequence of twisted Douglas - Rachford approximants is the sum of two sequences (u_n) and (v_n) resulting from the application of von Neumann's alternating projection algorithm to the pairs of subspaces M and N , and N^\perp and M^\perp respectively. Since the Friedrich angle θ between M and N is the same as the angle between M^\perp and N^\perp , the twisted Douglas - Rachford algorithm converges with the same rate as the von Neumann algorithm; namely at a linear rate proportional to $\cos^2 \theta$ [10].

THE SPECIAL CASE OF AN ELLIPSE AND A LINE. Without loss of generality we consider the ellipse $E : x^2 + \left(\frac{y}{b}\right)^2 = 1$, and the line $L : \alpha x + \beta y = \gamma$, where $b \geq 1$, $\beta \geq 0$ and to ensure the existence of $f = (x_0, y_0) \in L \cap E \cap \mathbb{R}^2$, either $\alpha \leq \gamma \leq \beta b$ or $\beta b < \gamma$ and $\alpha \geq \sqrt{\gamma^2 - \beta^2 b^2}$.

Following the strategy outlined above (see Figure -17) leads us to consider the

EIGENVALUES OF THE DOUGLAS - RACHFORD OPERATOR FOR TWO LINES THROUGH THE ORIGIN;

$$L_1 : \alpha x + \beta y = 0 \quad \text{and} \quad L_2 : Ax + By = 0$$

where in our context the latter line, being parallel to the tangent to E at f , has $A = x_0$ and $B = y_0/b^2$.

It is readily verified that the orthogonal projection onto L_1 has the matrix

$$[P_{L_1}] = \frac{1}{\alpha^2 + \beta^2} \begin{pmatrix} \beta^2 & -\alpha\beta \\ -\alpha\beta & \alpha^2 \end{pmatrix} \quad (17)$$

with a matching expression for $[P_{L_2}]$. Substituting these expressions into $T_{L_2 L_1} = 2P_{L_1} P_{L_2} - P_{L_1} - P_{L_2} + I$ yields

$$[T_{L_1 L_2}] = \frac{\psi}{\Delta} \begin{pmatrix} \psi & \omega \\ -\omega & \psi \end{pmatrix},$$

where $\psi = \alpha A + \beta B$, $\omega = \alpha B - \beta A$ and $\Delta = (\alpha^2 + \beta^2)(A^2 + B^2)$, which has

eigenvalues $\frac{\psi}{\Delta}(\psi \pm i\omega)$ with modulus squared equal to

$$\begin{aligned}\frac{\psi^2}{\Delta^2}(\psi^2 + \omega^2) &= \frac{(\alpha A + \beta B)^2((\alpha A + \beta B)^2 + (\alpha B - \beta A)^2)}{(\alpha^2 + \beta^2)^2(A^2 + B^2)^2} \\ &= \frac{(\alpha A + \beta B)^2}{(\alpha^2 + \beta^2)(A^2 + B^2)} \\ &< 1.\end{aligned}$$

Thus, as expected, for any two lines intersecting in a single point the Douglas - Rachford algorithm with any starting point spirals exponentially to their common point.

THEREFORE, the Douglas - Rachford algorithm for a line and an ellipse E is locally convergent at each of the feasible points f provided,

$$\|P_E(p) - P_{H_f}(p)\| = o(\|p - f\|),$$

for all p in some neighbourhood of f .

To see this we follow an argument suggested by Asen Dontchev [11]. While we present the argument in the particular case of the ellipse $E : x^2 + \left(\frac{y}{b}\right)^2 = 1$, the astute reader will observe that it applies to any smooth hypersurface $K : g(x) = 0$ at any point $f = (x_0, y_0) \in K$ at which the gradient ∇g is non-singular (true for the ellipse as $\nabla g(x) = (2x, 2y/b^2)$) and so applies to p -spheres except near the extreme points of the sphere when $0 < p \leq 1$.

We begin by noting that for the supporting hyperplane (tangent) to E at f

$$P_{H_f}(p) = f + P_{H_f-f}(p - f)$$

where

$$[P_{H_f-f}] = \frac{b^4}{b^4 x_0^2 + y_0^2} \begin{pmatrix} y_0^2/b^4 & -x_0 y_0/b^2 \\ -x_0 y_0/b^2 & x_0^2 \end{pmatrix}.$$

Next we observe that the nearest point projection $(u(p), v(p)) = P_E(p)$ at $p = (\zeta, \eta)$ is the solution of

$$\text{minimize: } \frac{1}{2}\|P_E(p) - p\|^2 = \frac{1}{2}((u - \zeta)^2 + (v - \eta)^2)$$

$$\text{subject to: } g(P_E(p)) = u^2 + \left(\frac{v}{b}\right)^2 - 1 = 0,$$

which, since $\nabla g(P_E(p))$ is non-singular, is characterised via the method of Lagrange multipliers by $\nabla \frac{1}{2}((u - \zeta)^2 + (v - \eta)^2) + \lambda \nabla g(P_E(p)) = 0$ together with $g(P_E(p)) = 0$, that is,

$$\begin{aligned}f_1 : u - \zeta + 2\lambda u &= 0 \\ f_2 : v - \eta + 2\lambda \frac{v}{b^2} &= 0 \\ g : u^2 + \left(\frac{v}{b}\right)^2 - 1 &= 0.\end{aligned}$$

ASIDE: This yields the implicit specification

$$P_E(\zeta, \eta) = \left(\frac{\zeta}{1 + 2\lambda}, \frac{b^2 \eta}{b^2 + 2\lambda} \right), \quad \text{where } \frac{\zeta^2}{(1 + 2\lambda)^2} + \frac{b^2 \eta^2}{(b^2 + 2\lambda)^2} = 1$$

mentioned elsewhere with λ replaced by $-t/2$.

In order to apply the implicit function theorem to ensure that u, v and λ are differentiable functions of ζ and η in a neighbourhood of f we require the Jacobian of the above system of equations with respect to the dependent variables, u, v and λ at $f, J(f)$, to be non-singular. Since $P_E(f) = f$ we see from the first (and second) equation that for $p = f$ the corresponding Lagrange multiplier is necessarily 0. Thus,

$$J(f) = \frac{\partial(f_1, f_2, g)}{\partial(u, v, \lambda)} \Big|_{(x_0, y_0, 0)} = \begin{pmatrix} 1 & 0 & 2x_0 \\ 0 & 1 & 2y_0/b^2 \\ 2x_0 & 2y_0/b^2 & 0 \end{pmatrix},$$

more generally $J(f) = \begin{pmatrix} I & \nabla g(f)^T \\ \nabla g(f) & 0 \end{pmatrix}$, which is indeed non-singular, and in our case

$$J(f)^{-1} = \frac{b^4}{b^4 x_0^2 + y_0^2} \begin{pmatrix} y_0^2/b^4 & -x_0 y_0/b^2 & x_0/2 \\ -x_0 y_0/b^2 & x_0^2 & y_0/2b^2 \\ x_0/2 & y_0/2b^2 & -1/4 \end{pmatrix}.$$

So the implicit function theorem applies to give

$$\begin{pmatrix} [P'_E(f)] \\ [\lambda'(f)] \end{pmatrix} = \frac{\partial(u, v, \lambda)}{\partial(\zeta, \eta)} \Big|_{(x_0, y_0)} = J(f)^{-1} \frac{\partial(f_1, f_2, g)}{\partial(\zeta, \eta)} = J(f)^{-1} \begin{pmatrix} I \\ 0 \end{pmatrix},$$

whence $[P'_E(f)] = \frac{b^4}{b^4 x_0^2 + y_0^2} \begin{pmatrix} y_0^2/b^4 & -x_0 y_0/b^2 \\ -x_0 y_0/b^2 & x_0^2 \end{pmatrix}$ which we recognise as $[P'_{H_f}(f)]$ and we are able to conclude that near f

$$\|P_E(p) - P_{H_f}(p)\| = \|f + P'_E(f)(p - f) + \Delta - (f + P'_{H_f}(f)(p - f))\| = \|\Delta\|$$

where $\|\Delta\| = o(\|p - f\|)$ as required.

So for an ellipse and generically intersecting line, the Douglas-Rachford algorithm is locally convergent at each of the feasible points.

5 Important Lessons about Global Behavior

What we have observed in our computer-assisted study of these two simple cases is remarkably informative: it suggests likely explanations for the behavior of the algorithm both for feasible and infeasible cases. We consider feasible cases first.

5.1 The Feasible Case

The authors of [1] experimented with using the Douglas Rachford method to solve Sudoku puzzles. From [1], the left hand images of Figures 18 and 19, illustrate the distance to the solution by iterations of Douglas Rachford for two different sudoku puzzles. First consider Figure 18. On the left, we see the algorithm struggle for a long period of time before finally converging. Compare this to the image on the right: for T_{E_2, L_2} with 210 iterates, distance of each iterate—to the particular feasible point the sequence converges to—is plotted.

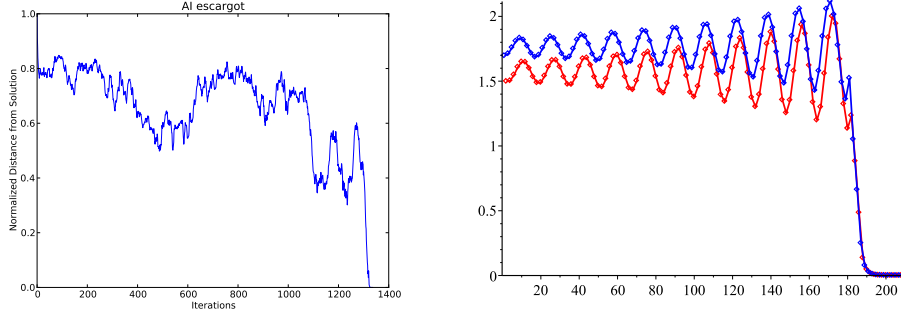


Figure 18: Left: from from [1] distance to the solution by iterations of Douglas Rachford for a sudoku puzzle. Right: for 210 iterates of T_{E_2, L_2} , distance from each iterate to the particular feasible point the sequence converges to.

The subsequences x_{2k} and x_{2k-1} are highlighted in red and blue respectively. They correspond respectively to iterates landing in the left and right source basins in Figures 11 and 16. Without the geometric intuition given from Figure 16, we would not know to color these two subsequences distinctly, and the plot of the sequence would appear less uniform than they do. Once the iterates have finally climbed free of the source basins, we see rapid convergence.

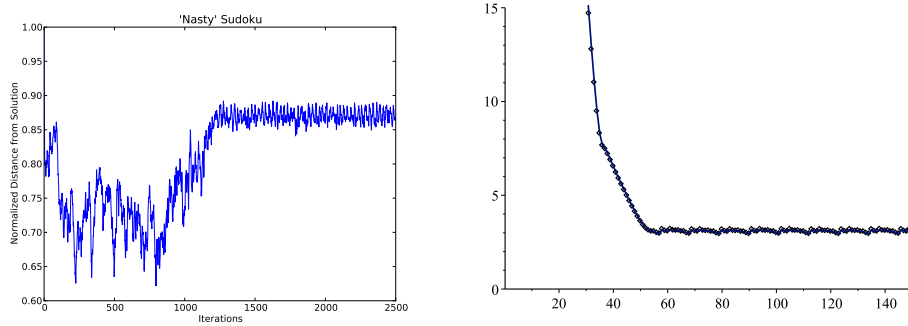


Figure 19: Left: from [7] distance to the solution by iterations of Douglas Rachford for a sudoku puzzle. Right: 150 iterates of T_{E_{14}, L_9} : The iterates approach the ellipse before being pulled into the attractive basins for period 11 points.

Now consider Figure 19. On the left, we see the the error for the Douglas Rachford iterates on another sudoku puzzle. This time the error stabilizes after a time without any indication of impending convergence. On the right, we see the iterates for T_{E_{14}, L_9} . The iterates approach the ellipse before being pulled into the attractive basins for some period 11 points, preventing convergence.

The authors of [2] and [7] — see also [3] — also experimented with using the Douglas Rachford method to solve matrix completion problems associated with incomplete euclidean distance matrices for protein mapping. Consider Figure 20. The left image shows the relative error of iterates when solving the Euclidean distance matrices for various proteins. The right image shows the

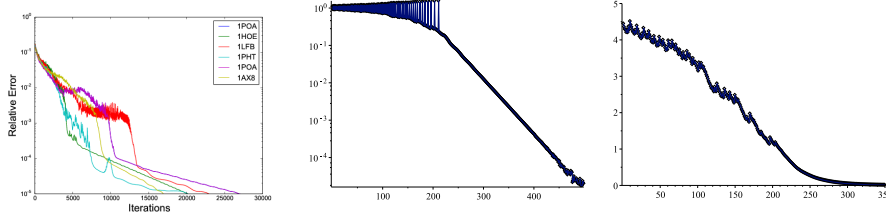


Figure 20: Left: from [7] relative error by iterations (Vertical axis logarithmic) for the Euclidean distance matrices for five proteins. Center: for T_{E_8, L_6} relative error by iterations (vertical axis logarithmic) for the 300 iterates which are pictured in Figure 21. Right: distance to the actual feasible point for the same 300 iterates.

absolute error for the iterates of T_{E_8, L_6} when the sequence of iterates is started near to basins of attraction. The center image shows the relative error. The

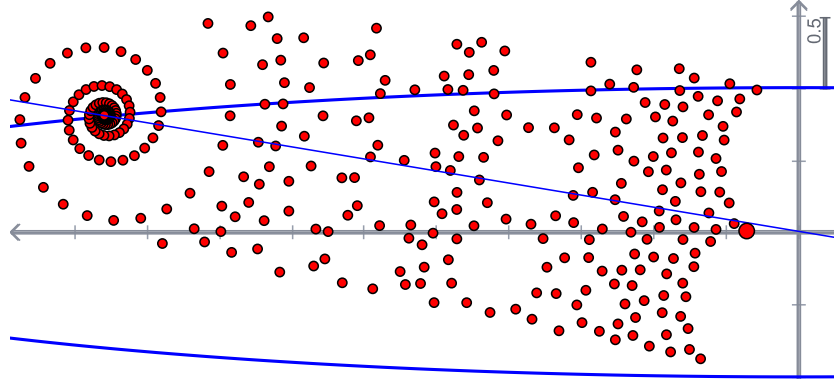


Figure 21: For T_{E_8, L_6} convergent sequences of iterates started among the basins of periodicity appear to trace out the shape of the basins on their way to the feasible point.

exact iterates used to generate this data are shown in Figure 21; they appear to trace out the shapes of the other attractive basins, narrowly dodging around them on their way to eventual convergence. Points started relatively close to basins of attraction for periodic points or inside of source basins (as in the right hand side of Figure 18) appear to take longer to converge than those started elsewhere.

For one of the larger proteins, convergence took an especially long time, and so the authors paused the computation after 5000 steps in order to render a picture of the protein. What they discovered was rather fascinating and is shown in Figure 22

At first the rendering on the right looks nothing like that of the true protein on the left. However, upon further scrutiny, it can be seen that there are actually two very good pieces which closely resemble the real protein and that they are merely separated by a great deal of “spaghetti.” Continuing to run the program, the relative error eventually dropped as in the left side of Figure 20. When the authors paused and rendered a picture of the protein after 28000 iterations,

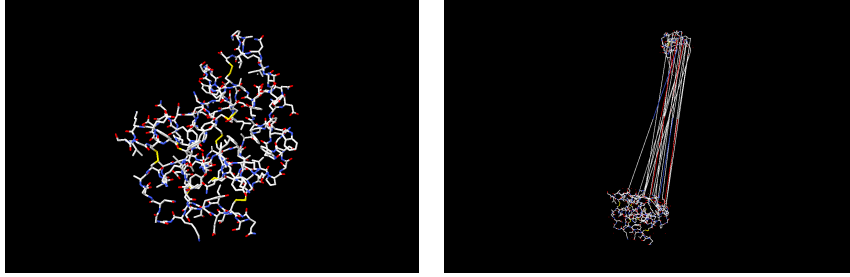


Figure 22: Left: the true protein 1POA. Right: the protein rendering from the Euclidean distance matrix problem paused after 5000 steps of Douglas-Rachford. Images from [2].

the resulting image was so difficult to tell apart from the true structure of the protein that we decline to include it here.

While we cannot say with any real certainty that the behavior when solving these Euclidean distance matrices is analogous to iterates climbing out from inside of repelling basins or dodging and weaving amongst attractive basins, it is surprising that a system as simple as that of a simple ellipse with a line can create behavior so similar to that in far more complicated systems.

We also extend our gratitude to the authors of [1],[2], and [7] for the use of their images.

5.2 The Infeasible Cases

For the infeasible cases of line and the p-sphere or ellipse, we observed that the iterates of the Douglas Rachford algorithm appear to walk to infinity with a roughly linear step size. In both infeasible cases, it is possible to strictly separate the two sets in question. This led to the following theorem.

Theorem 7. *In a Euclidean Space X , let A, B be sets. Further suppose one of the following:*

1. *A is compact and $\text{co}(A)$ and $\text{cl}(\text{co}(B))$ are disjoint.*
2. *B is compact and $\text{cl}(\text{co}(A))$ and $\text{co}(B)$ are disjoint.*

Then, where x_0 is the starting point and $\{x_n\}_{n=1}^{\infty}$ are the iterates for Douglas Rachford $T_{A,B}$, we have that $\|x_n\|$ tends linearly to ∞ with a step size of at least $d(A, B)$.

Proof. We can strictly separate $\text{co}(A)$ and $\text{cl}(\text{co}(B))$ with a hyperplane $H = f^{-1}(\alpha)$ for some linear functional f . See [9, Theorem 1.7] for details.

Because the Douglas Rachford Algorithm is invariant under a translation of the space, let $\alpha = 0$. Then H is a subspace, so we can uniquely describe any $x \in X$ as $x = h_x + y_x$ where $h_x \in H$ and $y_x \in H^{\perp}$. We can impose several additional properties on f :

$$|f(x)| = |f(h_x + y_x)| = \|y_x\|_X \quad \text{for all } x \in X \quad (18)$$

$$f(x) < 0 \quad \text{for all } x \in A \quad (19)$$

$$f(x) > 0 \quad \text{for all } x \in B. \quad (20)$$

With this set of conditions on f , it may help the reader to recognize that $|f(x)| = d(x, H)$ for all $x \in X$ while the sign of $f(x)$ indicates the “side” of H on which x lies. A negative sign corresponds to it lying on the “side” of set A while a positive sign corresponds to it lying on the “side” of set B . With this in mind, it should be clear that equations (18), (19), and (20) imply that

$$f(x) \geq d(A, B) \quad \text{for all } x \in B - A. \quad (21)$$

Straight from the construction of the Douglas Rachford function we have that

$$x_{n+1} - x_n = \frac{R_B(R_A(x_n)) + x_n}{2} - x_n = \frac{R_B(P_A(x_n)) - x_n}{2}.$$

From the definition of the reflection R_B , the RHS expands to

$$\frac{2P_B(R_A(x_n)) - R_A(x_n) - x_n}{2}. \quad (22)$$

From the definition of the reflection R_A , this expands to

$$\frac{2P_B(R_A(x_n)) - (2P_A(x_n) - x_n) - x_n}{2} = P_B(R_A(x_n)) - P_A(x_n) \in B - A. \quad (23)$$

Now $x_{n+1} - x_n \in B - A$ implies that $f(x_{n+1} - x_n) \geq d(A, B)$. Thus we have that, for all n ,

$$f(x_{n+1}) \geq d(A, B) + f(x_n).$$

This shows that $\|x_n\|_n \rightarrow \infty$ with a linear step size of at least $d(A, B)$. \square

From this result we obtain the following corollary, the computer-assisted discovery of which motivated the pursuit of the more general Theorem.

Corollary 8. *Where the n th iterate of the Douglas Rachford algorithm in the infeasible case of a line L with an ellipse E or a p -sphere S is denoted x_n , we have that $\|x_n\| \rightarrow \infty$ with a linear step size greater than or equal to $d(E, L)$ or $d(E, S)$ respectively.*

Using these result and the following remark, we can naturally extend some of the convex theory to the non-convex case.

Remark 2. The authors of [4, Theorem 5.5] showed that for convex subsets A, B of a real Hilbert space X , where $v = P_{\text{cl}(\text{ran}(\text{Id} - T_{U,V}))}$, $x \in X$, and $U \cap (v + V) \neq \emptyset$, we have that $(P_U T_{U,V}^n x)_n$ converges weakly to a point in $U \cap (v + V)$.

We extend this result using Theorem 7.

Theorem 9. *Let A, B be the boundaries of closed, convex sets U, V respectively in a Euclidean space X where one of them is compact and the intersection of U, V is empty. Let $v = P_{\text{cl}(\text{ran}(\text{Id} - T_{U,V}))}$ be the uniquely defined element in $\text{cl}(\text{ran}(\text{Id} - T_{U,V}))$ such that $\|v\| = \inf_{x \in X} \|x - Tx\|$. Let $v' = P_{\text{cl}(\text{ran}(\text{Id} - T_{A,B}))}$ be the uniquely defined element in $\text{cl}(\text{ran}(\text{Id} - T_{A,B}))$ such that $\|v'\| = \inf_{x \in X} \|x - Tx\|$. Then, where $x \in X$, we have that $(P_A T_{A,B}^n x)_n$ converges weakly to a point in $A \cap (v' + B)$.*

Proof. By the closure of both sets and compactness of one of the sets, we have attainment of elements which minimize the distance. Thus $A \cap (v' + B) = U \cap (v + V) \neq \emptyset$.

Let f be defined as in Theorem 7. Then the sequence $f(x_n)$ is monotone increasing and so there exists some $n' \in \mathbb{N}$ such that $f(x_n) \geq 0$ for all $n \geq n'$. Suppose $n \geq n'$. Then we have that $x_n \notin U$. Thus

$$P_A(x_n) = P_U(x_n). \quad (24)$$

Since the projection is the same, so also will the reflection be:

$$R_A(x_n) = R_U(A)(x_n). \quad (25)$$

We also have that $R_A(x_n) \notin V$. For this reason we have that

$$P_B(P_A(x_n)) = P_V(R_A(x_n)) \quad (26)$$

and so we have the reflection equality

$$R_B(R_A(x_n)) = R_V(R_A(x_n)). \quad (27)$$

Thus we have that

$$T_{A,B}(x_n) = T_{U,V}(x_n). \quad (28)$$

Thus we have that

$$(P_A T^{n'+n} x)_n = (P_A T^n x_{n'})_n = (P_A T_{U,V}^n x_{n'})_n \quad (29)$$

for all $n \in \mathbb{N}$. We have from Remark 2 that the sequence on the right converges weakly to a point y in $U \cap (v + V)$. \square

5.3 Closing Remarks

Given that we are studying non-convex Douglas Rachford using some of the simplest sets possible, it is remarkable that such extraordinary complexity appears. More interesting from a technical standpoint is the similarity with which the behavior on such simple sets appears to resemble some of the behavior observed on much larger and more complicated ones. We also hope that we have succeeded in making the case for computer-assisted discovery, visualization, and verification.

References

- [1] F.J. Aragón Artacho, J.M. Borwein, and M.K. Tam. “Recent results on Douglas-Rachford methods for combinatorial optimization problems,” *Journal of Optimization Theory and Applications*, **163** (2014) 1-30.
- [2] F.J. Aragón Artacho, J.M. Borwein, M.K. Tam. “Douglas-Rachford feasibility methods for matrix completion problems,” *ANZIAM Journal*. **55**(4) (2014) 299-326.

- [3] David H. Bailey and Jonathan M. Borwein, “Experimental computation as an ontological game changer: The impact of modern mathematical computation tools on the ontology of mathematics.” In *Mathematics, Substance and Surmise: Views on the Meaning and Ontology of Mathematics*, Springer, 25-67.
- [4] Heinz H. Bauschke and Walaa M. Moursi, “On the Douglas-Rachford algorithm,” <http://arxiv.org/pdx/1604.04603.pdf>.
- [5] J.M. Borwein. “The Life of Modern Homo Habilis Mathematicus: Experimental Computation and Visual Theorems.” *Tools and Mathematics*, 23-90, in *Mathematics Education Library*, vol. 347, Springer, 2016.
- [6] J.M. Borwein and B. Sims, The Douglas-Rachford algorithm in the absence of convexity, *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, Springer Optimization and its Applications: 49 (2011), 93-109.
- [7] J.M. Borwein and Matthew K. Tam. “Reflection methods for inverse problems with applications to protein conformation determination,” Springer volume on the CIMPA school [?, ?], New Delhi, India, Dec. 2012.
- [8] J. Benoist, “The Douglas-Rachford Algorithm for the Case of the Sphere and the Line,” *Journal of Global Optimization*: 63 (2015), 363-380.
- [9] Haim Brezis, *Functional Analysis, Sobolev Spaces and Partial Differential Equations*, Springer (2011).
- [10] F. Deutsch, Rate of convergence of the method of alternating projections, in B. Brosowski and F. Deutsch (eds), *Parametric Optimization and Approximation*, Birkhäuser, Basel, 1983, pp96–107.
- [11] Asen L. Dontchev, private communication.
- [12] J. Douglas and H.H. Rachford, “On the numerical solution of the heat conduction problem in 2 and 3 space variables,” *Transactions of the AMS* **82** (1956), 421–439.
- [13] V. Lakshmikantham and D. Trigiante, *Theory of Difference Equations - Numerical Methods and Applications*, Marcel Dekker, 2002, ppx+300.
- [14] P.L. Lions and B. Mercier, “Splitting algorithms for the sum of two nonlinear operators,” *SIAM Journal on Numerical Analysis*, **16** (1979), 964–979.
- [15] Tristan Needham, *Visual complex analysis*, Clarendon Press, Oxford, 1997, ppxiii+612.

6 Appendix: Proofs and Formulae

Claim 10. *Proposition (2) holds.*

Proof. Since $L = \{(x, y) \in \mathbb{R}^2 | y - x = 0\}$ is a subspace the projection P_L is linear and by equation (17)

$$[P_L] = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}.$$

So, by equation (15)

$$T(x) := T_{AL}(x) = x - P_L(x) - P_A(x) + 2P_L P_A(x).$$

Thus,

$$\begin{aligned} T^2(x) &= x - P_L(x) - P_A(x) + 2P_L P_A(x) - P_L(x - P_L(x) - P_A(x) + 2P_L P_A(x)) \\ &\quad - P_A(T(x)) + 2P_L P_A(T(x)) \\ &= x - P_L(x) - P_A(x) + P_L P_A(x) - P_A(T(x)) + 2P_L P_A(T(x)), \end{aligned}$$

where we have made use of the fact that P_L is both linear and idempotent. So, requiring x to be a period 2 point, that is $T^2(x) = x$, is equivalent to requiring

$$(I - 2P_L)P_A(T(x)) = -P_L(x) - P_A(x) + P_L P_A(x).$$

Observing that $I - 2P_L$ is self-inverse, this is in turn equivalent to

$$\begin{aligned} P_A(T(x)) &= (I - 2P_L)(P_L(P_A(x) - x) - P_A(x)) \\ &= P_L(x) + (P_L - I)P_A(x). \end{aligned}$$

Converting to a matrix equation and using the specific form of $[P_L]$ in our case we have

$$[P_A(T(x))] = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \frac{1}{2} \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} P_A(x)_1 \\ P_A(x)_2 \end{pmatrix}.$$

Thus, $T(x) = x$ if and only if

$$P_A(T(x)) = \frac{1}{2}(x_1 + x_2)(1, 1) + \frac{1}{2}(P_A(x)_1, P_A(x)_2)$$

As claimed. \square

Remark 3. The closed forms for period 2 points (u, v) as in Proposition 6 are

as follows.

$$\begin{aligned}
u = & \pm 3\sqrt{2}((64 - 4\sqrt{Q}b^2 + \sqrt{Q}m^6 - 12\sqrt{Q}m^2 - 30\sqrt{Q}b^2m^4 \\
& + 9\sqrt{Q}b^4m^2 + 36\sqrt{Q}b^4m^4 - 4\sqrt{Q}b^2m^6 + 24\sqrt{Q}b^2m^2 + m^{10} \\
& + 144b^6m^8 + 16b^4m^{10} + 216b^6m^6 - 8b^2m^{10} + 81b^6m^4 + 9b^6m^2 \\
& - 112b^4m^8 - 77b^4m^6 + 35b^2m^8 + 7b^4m^4 + 67b^2m^6 - 40b^4m^2 + 5m^8 \\
& - 8m^6 + 4b^4 - 44m^4 + 152b^2m^2 + 208b^2m^4 + 32m^2 - 32b^2 + 16\sqrt{Q}) \\
& \cdot (m^2 + 1))^{\frac{1}{2}}(2b^2m^2 - b^2 - m^2 + 2)mb/(64 - 4\sqrt{Q}b^2 + \sqrt{Q}m^6 \\
& - 12\sqrt{Q}m^2 - 30\sqrt{Q}b^2m^4 + 9\sqrt{Q}b^4m^2 + 36\sqrt{Q}b^4m^4 - 4\sqrt{Q}b^2m^6 \\
& + 24\sqrt{Q}b^2m^2 + m^{10} + 144b^6m^8 + 16b^4m^{10} + 216b^6m^6 - 8b^2m^{10} \\
& + 81b^6m^4 + 9b^6m^2 - 112b^4m^8 - 77b^4m^6 + 35b^2m^8 + 7b^4m^4 + 67b^2m^6 \\
& - 40b^4m^2 + 5m^8 - 8m^6 + 4b^4 - 44m^4 + 152b^2m^2 + 208b^2m^4 + 32m^2 \\
& - 32b^2 + 16\sqrt{Q}) \\
v = & - \left(-4b^2m^4 + 13b^2m^2 + m^4 - b^2 - 13m^2 + (16b^4m^8 + 40b^4m^6 \right. \\
& - 8b^2m^8 + 33b^4m^4 - 14b^2m^6 + m^8 + 10b^4m^2 - 12b^2m^4 + 10m^6 \\
& + b^4 - 14b^2m^2 + 33m^4 - 8b^2 + 40m^2 + 16)^{\frac{1}{2}} + 4 \Big) u/6m(2b^2m^2 \\
& - b^2 - m^2 + 2)
\end{aligned} \tag{30}$$

where Q is given by

$$\begin{aligned}
Q = & (m^2 + 1)^2 (4b^2m^2 + 4bm^2 + b^2 + m^2 + 4b + 4) (4b^2m^2 - 4bm^2 \\
& + b^2 + m^2 - 4b + 4).
\end{aligned} \tag{32}$$

Remark 4. In Remark 5, $T(u, v) = (x, y)$ where

$$\begin{aligned}
0 = & \frac{1}{16m^2(-m^2x + 2my + Q + x)^2} x^4m^8 - 4x^3m^8u + 6x^2m^8u^2 \\
& - 4xm^8u^3 + m^8u^4 + 4x^4m^6 - 12x^3m^6u - 8x^2y^2m^6 \\
& + 16x^2m^6u^2 + 16xy^2m^6u - 12xm^6u^3 - 8y^2m^6u^2 + 4m^6u^4 \\
& - 8Qx^2ym^5 + 16Qxym^5u - 8Qym^5u^2 - 16x^3ym^5 \\
& + 40x^2ym^5u - 16xym^5u^2 - 8ym^5u^3 - 2Q^2x^2m^4 + 4Q^2xm^4u \\
& - 2Q^2m^4u^2 - 8Qx^3m^4 + 20Qx^2m^4u - 8Qxm^4u^2 - 4Qm^4u^3 \\
& - 2x^4m^4 + 12x^3m^4u - 16x^2y^2m^4 - 16x^2m^6 + 4x^2m^4u^2 \\
& + 16xy^2m^4u - 20xm^4u^3 + 16y^4m^4 + 16y^2m^4u^2 + 6m^4u^4 \\
& - 16Qx^2ym^3 + 16Qxym^3u + 32Qy^3m^3 + 16Qym^3u^2 - 32x^3ym^3 \\
& + 48x^2ym^3u + 64xy^3m^3 + 64xym^5 + 32xym^3u^2 - 32y^3m^3u \\
& - 16ym^3u^3 - 4Q^2x^2m^2 + 4Q^2xm^2u + 24Q^2y^2m^2 + 4Q^2m^2u^2 \\
& - 16Qx^3m^2 + 24Qx^2m^2u + 96Qxy^2m^2 + 32Qxm^4 + 16Qxm^2u^2 \\
& - 48Qy^2m^2u - 8Qm^2u^3 - 12x^4m^2 + 28x^3m^2u + 88x^2y^2m^2 \\
& + 32x^2m^4 + 16x^2m^2u^2 - 96xy^2m^2u - 20xm^2u^3 - 64y^2m^4 \\
& + 24y^2m^2u^2 + 4m^2u^4 + 8Q^3ym + 48Q^2xym - 24Q^2ymu \\
& + 88Qx^2ym - 96Qxymu - 64Qym^3 + 24Qymu^2 + 48x^3ym \\
& - 88x^2ymu - 64xym^3 + 48xymu^2 - 8ymu^3 + Q^4 + 8Q^3x \\
& - 4Q^3u + 22Q^2x^2 - 24Q^2xu - 16Q^2m^2 + 6Q^2u^2 + 24Qx^3 \\
& - 44Qx^2u - 32Qxm^2 + 24Qxu^2 - 4Qu^3 + 9x^4 - 24x^3u \\
& - 16x^2m^2 + 22x^2u^2 - 8xu^3 + u^4
\end{aligned} \tag{33}$$

$$\begin{aligned}
v = & \frac{1}{2m} (m^2u - m^2x + 2my - u + x + (x^2m^4 - 2xm^4u + m^4u^2 \\
& - 4xym^3 - 2x^2m^2 - 4xm^2u + 4y^2m^2 + 2m^2u^2 + 4xym + x^2 \\
& - 2xu + u^2)^{\frac{1}{2}})
\end{aligned} \tag{34}$$

where Q is defined by

$$\begin{aligned}
Q = & (x^2m^4 - 2xm^4u + m^4u^2 - 4xym^3 - 2x^2m^2 - 4xm^2u \\
& + 4y^2m^2 + 2m^2u^2 + 4xym + x^2 - 2xu + u^2)^{\frac{1}{2}}
\end{aligned} \tag{35}$$

7 Appendix: Numerical Projections

The use of built-in function solvers proved both slow and insufficiently accurate for the task of computing projections. The causes for this were twofold:

- The shapes and properties of the functions induced by solving the Lagrangian problem - for most situations - render Newton's Method alone insufficient.
- We encounter diminishing returns on increased accuracy from the function solver when solving for one variable in cases where small inaccuracies in

that variable corresponded to much larger inaccuracies in the variable not being solved for.

We compensate for these obstacles in different ways for the p -sphere and ellipse problems. To create a projection protocol which works for any point, it suffices - by symmetry - to consider only projections of points in the first quadrant. Points in other quadrants may be reflected into the first quadrant and then their first quadrant projections reflected back into the appropriate quadrant. In each case we consider the computation of the projection (x, y) of a point (u, v) onto the set in question.

7.1 Projection for the p -Sphere

For $p > 1$, an error in x corresponds to a larger error in y for any (u, v) below the line $y = x$ while an error in x corresponds to a smaller error in y for (u, v) above the line $y = x$. This is clear from the geometry.

The opposite relationship is true for $p < 1$. For $p < 1$, an error in x corresponds to a smaller error in y for any (u, v) below the line $y = x$ while an error in x corresponds to a larger error in y for (u, v) above the line $y = x$.

By symmetry, we may reflect a point across the line $y = x$, compute the projection for its reflection, and then obtain the projection for the point by again reflecting across $y = x$.

For $p > 1$, we exploit this symmetry by reflecting points to the region above the line $y = x$ so that an error in x always corresponds to a smaller error in y . Likewise, for $p < 1$, we exploit this symmetry by reflecting points to the region below the line $y = x$ so that an error in x always corresponds to a smaller error in y . This inexpensive if-then protocol removes the obstacle of diminishing returns in accuracy when using Newton's method. We henceforth consider the projection after reflecting to the appropriate region.

Because of this protocol, we may begin our search for the root on a smaller interval of the x -axis. For $p > 1$, the point of projection will be to the left of the intersection of $y = x$ and the p -sphere. For $p < 1$, it will be to the right. For $p > 1$, the interval is $[0, \exp(-\log(2)/p)]$; for $p < 1$, the interval is $[\exp(-\log(2)/p), 1]$.

For $p > 1$, Newton's method is sufficient and converges rapidly. For $p < 1$, the shape of the functions necessitates the use of bisection method prior to implementing Newton's method.

The complete routine is as follows:

1. Global Projection Function

- (a) Takes as input a point and reflects it to Quadrant I
- (b) Sends this point through Quadrant Projection Function
- (c) Reflects the output from the Quadrant Projection Function back to the appropriate quadrant and returns this value.

2. Quadrant Projection Function

- (a) Take as input a point (u, v) in the first quadrant.
- (b) Case: $p > 1$

- i. Case: (u, v) below the line $y = x$.
 - A. Reflect (u, v) across the line $y = x$.
 - B. Send the reflection through High Projection Function
 - C. Reflect the output across the line $y = x$ and return this value.
 - ii. Case: (u, v) above the line $y = x$. Send point through High Projection Function and return output.
 - (c) Case: $p < 1$
 - i. Case: (u, v) below the line $y = x$. Send point through Low Projection Function and return output.
 - ii. Case: (u, v) above the line $y = x$.
 - A. Reflect (u, v) across the line $y = x$.
 - B. Send the reflection through Low Projection Function
 - C. Reflect the output across the line $y = x$ and return this value.
- 3. High Projection Function
 - (a) Take in a point (u, v)
 - (b) Perform Newton's Method on resultant Lagrangian function to within a specified error threshold to compute the x value.
 - (c) Compute the y value corresponding to the x value.
 - (d) Return the value (x, y)
- 4. Low Projection Function
 - (a) Take in a point (u, v)
 - (b) Perform bisection method on resultant Lagrangian function to compute the x value to an error threshold of .01.
 - (c) Take the x value from the bisection step and continue with Newton's method to compute x to within a specified error threshold.
 - (d) Compute the y value corresponding to the x value.
 - (e) Return the value (x, y)

This routine performs relatively quickly in Cinderella, computing 256,000 Douglas Rachford iterates in about 25 seconds.¹

7.2 Projection for the Ellipse

For the Ellipse, we obtain a similar problem with diminishing returns. For points of projection nearer to $x = 1$, an error in x corresponds to a larger error in y . For points of projection nearer to $y = 1$, an error in y corresponds to a larger error in x . However, we lose the symmetry across the line $y = x$ that we had in the case of the p -sphere.

The natural temptation is to compute the value (w, z) at which the curve of the ellipse has tangent slope -1 and then use one Newton solver to compute the y value of the projection for points (u, v) below the line $y = \frac{z}{w}x$ and a different

¹Device: Dell Latitude E7250

Newton solver to compute the x value of the projection for points (u, v) above it.

Worthy of note is that although the Lagrangian functions differ, the interval over which we search for a root is $[0, 1]$ in both the case where we are solving for x and the case where we are solving for y . In the case where we are solving for y , we obtain the y value by multiplying the root by b .

However, the for $b < 1$, the induced Lagrangian function to solve for x has multiple roots in $[0, 1]$. Symmetrically, for $b > 1$, the Lagrangian function to solve for y has multiple roots in $[0, 1]$. For this reason, such a method is not reliable.

Fortunately, for $b < 1$, the induced Lagrangian function to solve for y has a single root in $[0, 1]$. Symmetrically, for $b > 1$, the Lagrangian function to solve for x has only one root in the interval $[0, 1]$. Thus we can compute the projection by choosing the appropriate method depending only on the value of b .

In either case, depending upon the (u, v) point we seek to project, there may be other roots outside of but very near to $[0, 1]$. For this reason, we must still use bisection method prior to Newton's method. The complete routine is as follows.

1. Global Projection Function

- (a) Takes as input a point and reflects it to Quadrant I
- (b) Sends this point through Quadrant Projection Function
- (c) Reflects the output from the Quadrant Projection Function back to the appropriate quadrant and returns this value.

2. Quadrant Projection Function

- (a) Case: $b > 1$
 - i. Take in a point (u, v)
 - ii. Perform five steps of bisection method on resultant Lagrangian function for x to compute a value near the root.
 - iii. Take the final value from the bisection step and continue with Newton's method to compute x to within a specified error threshold.
 - iv. Compute the y value corresponding to the x value.
 - v. Return the value (x, y)
- (b) Case: $b \leq 1$
 - i. Take in a point (u, v)
 - ii. Perform five steps of bisection method on resultant Lagrangian function for y to compute a value near the root.
 - iii. Take the final value from the bisection step and continue with Newton's method to compute to within a specified error threshold for the root. Multiply by b to obtain the y value.
 - iv. Compute the x value corresponding to the y value.
 - v. Return the value (x, y)

This routine performs relatively quickly in Cinderella, computing 253,128 Douglas Rachford iterates in about 23 seconds.².

7.3 Parallelized Code

Matt's section here

8 Appendix: Numerical Inversion

In order to build the curves numerically, we would need to start at a point (x, y) and find a corresponding point (u, v) such that $T(u, v) = (x, y)$. The difference vector $(x, y) - (u, v)$ is tangent to the level curve at (x, y) . We would approximate the level curve by normalizing the difference vector, shrinking it to a length ϵ , plotting it with its origin at (x, y) , and repeating the process beginning at its endpoint. Critically, this process requires numerical inversion of the Douglas Rachford algorithm. Armed with the the closed form for a Lyapunov function from the 2-sphere from [8], we first tested numerical inversion on the 2-sphere. We discovered the implementation to be problematic.

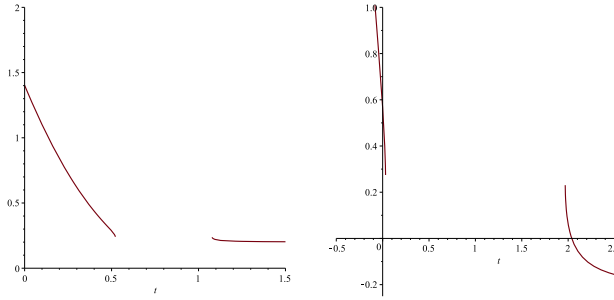


Figure 23: Left: the function resulting from the right side of Equation (33) in the case $(x, y) = (.8, .75)$ which is not in the image of T . Right: the function resulting from the right side of Equation (33) in the case $(x, y) = (1, 0.25) = T([2.039402300, 0.6279644710])$. *Maple's* built-in *fsolve* command fails to find the root which had to be found with the *NextZero* command - in the *RootFinding* package - with the manual specification that the root is greater than 2.

Remark 5. For T_{S_2, L_m} with $m \neq 0$, given a point $x_1 = (x, y)$ such that $\|x_1\| \leq 1$, we can numerically approximate $x_0 = (u, v)$ such that $Tx_0 = x_1$. The method is found in Remark 4 in Appendix 6 where the function to be solved is in Equation (33). *Maple* appears to solve this function easily when the condition $\|x_1\| \leq 1$ is satisfied.

Proof. Let $P_S(x_0) = (w, z)$. We obtain the result by solving for v and then u in the system

$$\begin{aligned} u - w &= \lambda 2w, & v - z &= \lambda \cdot 2z \\ w^2 + z^2 &= 1 \end{aligned}$$

²Device: Dell Latitude E7250

where the equations simply come from the Lagrangian system. *Maple* yields two possibilities for v . In cases where there are two possible values (u, v) such that $T(u, v) = (x, y)$, we find that using the second value for v instead of the first value merely changes the order in which *Maple* finds the values for u (the roots of Equation (33)) and so changes the order in which we find the two possible (u, v) values. In cases where there is only one possible value (u, v) , we find that the first selection for v (reported in Equation (34)) gives us the correct solution. Even in the event where there are two possible (u, v) values, we only need one for the sake of building numerical Lyapunov level curves because the two difference vectors will be colinear and we can always choose the more appropriate direction to step in by comparing to the direction of the previous step. \square

In the case where $x^2 + y^2 \geq 1$, Equation (33) often behaves in an undesirable way, inhibiting normal numerical solving. Cases wherein no root could be seen in the graphed function coincided with cases wherein (x, y) was not actually in the image of T . Both of these phenomena are illustrated in Figure 23, and the latter phenomenon is, of course, in keeping with what we would expect. For this reason, numerical inversion will obviously not work for plotting Lyapunov curves in most regions outside of the 2-sphere and any ellipse.

While it seemed that tinkering with the root search might still yield some hope for using numerical inversion on the 2-sphere, at least for producing partial Lyapunov curves in areas mapped onto by T , attempts to extend a numerical problem to the ellipse with $b \neq 1$ proved unsuccessful.

9 Appendix: Other Images

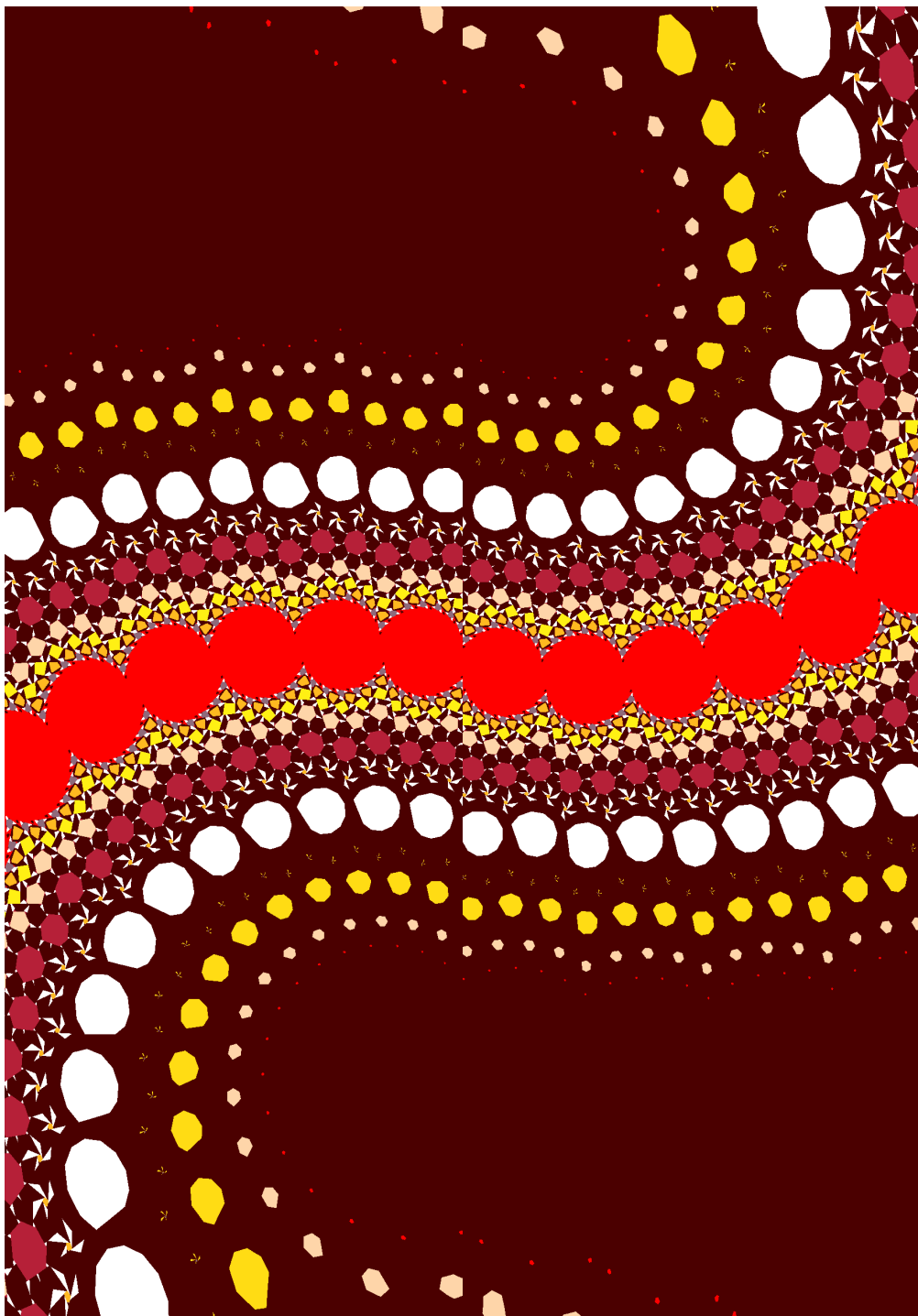


Figure 24: A rendering of basins of attraction for T_{E_8, L_6} with colors based on indigenous Australian art.